

VSI Archive Backup System for OpenVMS MDMS Reference Guide

Operating System and Version: VSI OpenVMS x86-64 Version V9.2-3 or higher
VSI OpenVMS IA-64 Version 8.4-1H1 or higher
VSI OpenVMS Alpha Version 8.4-2L1 or higher

Software Version: ABS/MDMS V4.x

VSI Archive Backup System for OpenVMS MDMS Reference Guide



VMS Software

Copyright © 2026 VMS Software, Inc. (VSI), Boston, Massachusetts, USA

Legal Notice

Confidential computer software. Valid license from VSI required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

The information contained herein is subject to change without notice. The only warranties for VSI products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. VSI shall not be liable for technical or editorial errors or omissions contained herein.

All other trademarks and registered trademarks mentioned in this document are the property of their respective holders.

Table of Contents

Preface	vii
1. About VSI	vii
2. Intended Audience	vii
3. OpenVMS Documentation	vii
4. VSI Encourages Your Comments	vii
5. Conventions	vii
Chapter 1. MDMS DCL Commands	1
MDMS ALLOCATE DRIVE	1
MDMS ALLOCATE VOLUME	5
MDMS BIND VOLUME	10
MDMS CANCEL REQUEST	11
MDMS CREATE ARCHIVE	13
MDMS CREATE CATALOG	18
MDMS CREATE DRIVE	21
MDMS CREATE ENVIRONMENT	26
MDMS CREATE GROUP	32
MDMS CREATE JUKEBOX	35
MDMS CREATE LOCATION	41
MDMS CREATE MAGAZINE	43
MDMS CREATE MEDIA_TYPE	47
MDMS CREATE NODE	49
MDMS CREATE POOL	54
MDMS CREATE RESTORE	57
MDMS CREATE SAVE	67
MDMS CREATE SCHEDULE	75
MDMS CREATE SELECTION	80
MDMS CREATE VOLUME	84
MDMS DEALLOCATE DRIVE	93
MDMS DEALLOCATE VOLUME	94
MDMS DELETE ARCHIVE	96
MDMS DELETE CATALOG	97
MDMS DELETE DRIVE	98
MDMS DELETE ENVIRONMENT	99
MDMS DELETE GROUP	100
MDMS DELETE JUKEBOX	101
MDMS DELETE LOCATION	102
MDMS DELETE MAGAZINE	102
MDMS DELETE MEDIA_TYPE	103
MDMS DELETE NODE	104
MDMS DELETE POOL	105
MDMS DELETE RESTORE	106
MDMS DELETE SAVE	107
MDMS DELETE SCHEDULE	108
MDMS DELETE SELECTION	109
MDMS DELETE VOLUME	110
MDMS INITIALIZE VOLUME	111
MDMS INVENTORY JUKEBOX	113
MDMS LOAD DRIVE	116
MDMS LOAD VOLUME	118

MDMS MOVE MAGAZINE	121
MDMS MOVE VOLUME	124
MDMS REPORT VOLUME	128
MDMS SET ARCHIVE	133
MDMS SET CATALOG	138
MDMS SET DOMAIN	140
MDMS SET DRIVE	146
MDMS SET ENVIRONMENT	150
MDMS SET GROUP	156
MDMS SET JUKEBOX	158
MDMS SET LOCATION	162
MDMS SET MAGAZINE	164
MDMS SET MEDIA_TYPE	168
MDMS SET NODE	170
MDMS SET POOL	174
MDMS SET RESTORE	176
MDMS SET SAVE	185
MDMS SET SCHEDULE	192
MDMS SET SELECTION	197
MDMS SET SERVER	201
MDMS SET VOLUME	202
MDMS SHOW ARCHIVE	211
MDMS SHOW CATALOG	212
MDMS SHOW DOMAIN	217
MDMS SHOW DRIVE	218
MDMS SHOW ENVIRONMENT	220
MDMS SHOW GROUP	222
MDMS SHOW JUKEBOX	224
MDMS SHOW LOCATION	225
MDMS SHOW MAGAZINE	227
MDMS SHOW MEDIA_TYPE	229
MDMS SHOW NODE	230
MDMS SHOW POOL	232
MDMS SHOW REQUEST	233
MDMS SHOW RESTORE	235
MDMS SHOW SAVE	237
MDMS SHOW SCHEDULE	239
MDMS SHOW SELECTION	241
MDMS SHOW SERVER	242
MDMS SHOW VERSION	244
MDMS SHOW VOLUME	245
Start_Session	248
MDMS SYNCHRONIZE RESTORE	249
MDMS SYNCHRONIZE SAVE	250
MDMS UNBIND VOLUME	252
MDMS UNLOAD DRIVE	253
MDMS UNLOAD VOLUME	254
Chapter 2. ABS to MDMS Command Mappings	257
2.1. ABS CREATE CATALOG	257
2.2. ABS CREATE ENVIRONMENT	257
2.3. ABS CREATE STORAGE_CLASS	258
2.4. ABS DELETE	259

2.5. ABS DELETE CATALOG	259
2.6. ABS LOOKUP	260
2.7. ABS REPORT SAVE_LOG	260
2.8. ABS RESTORE	261
2.9. ABS SAVE	262
2.10. ABS SET CATALOG	263
2.11. ABS SET ENVIRONMENT	264
2.12. ABS SET RESTORE	265
2.13. ABS SET SAVE	266
2.14. ABS SET STORAGE_CLASS	267
2.15. ABS SHOW CATALOG	268
2.16. ABS SHOW ENVIRONMENT	268
2.17. ABS SHOW REQUEST_STATUS	269
2.18. ABS SHOW RESTORE	269
2.19. ABS SHOW SAVE	269
2.20. ABS SHOW STORAGE_CLASS	270
2.21. ABS SYNCHRONIZE	270
Chapter 3. MDMS Rights and Privileges	271
3.1. Types of MDMS Rights	271
3.1.1. High-Level Rights	271
3.1.2. Low-Level Rights	272
3.1.3. ABS Rights	274
3.2. Default High-Level to Low-Level Mapping	274
3.2.1. MDMS_USER Rights	274
3.2.2. MDMS_OPERATOR Rights	275
3.2.2.1. Domain Commands for Mapping Privileges	275
Chapter 4. MDMS Logical Names	277
Chapter 5. ABS Logical Names	281
5.1. ABS Logical Names	281
5.2. DEBUG Logicals	281
5.3. Miscellaneous Logicals	282
Chapter 6. MDMS Messages	285
Chapter 7. ABS Messages	329

Preface

1. About VSI

VMS Software, Inc. (VSI) is an independent software company licensed by Hewlett Packard Enterprise to develop and support the OpenVMS operating system.

2. Intended Audience

The intended audience is the system administrators who install the Disk File Optimizer software.

3. OpenVMS Documentation

The full VSI OpenVMS documentation set can be found on the VMS Software Documentation webpage at <https://docs.vmssoftware.com>.

4. VSI Encourages Your Comments

You may send comments or suggestions regarding this manual or any VSI document by sending electronic mail to the following Internet address: <docinfo@vmssoftware.com>. Users who have VSI OpenVMS support contracts through VSI can contact <support@vmssoftware.com> for help with this product.

5. Conventions

The following conventions are used in this manual:

Convention	Meaning
Ctrl/x	A sequence such as Ctrl/x indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.
...	A horizontal ellipsis in examples indicates one of the following possibilities: • Additional optional arguments in a statement have been omitted. • The preceding item or items can be repeated one or more times. • Additional parameters, values, or other information can be entered.
⋮	A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.
()	In command format descriptions, parentheses indicate that you must enclose the options in parentheses if you choose more than one.
[]	In command format descriptions, brackets indicate optional choices. You can choose one or more items or no items. Do not type the brackets on the command line. However, you must include the brackets in the syntax for OpenVMS directory specifications and for a substring specification in an assignment statement.

Convention	Meaning
	In command format descriptions, vertical bars separate choices within brackets or braces. Within brackets, the choices are options; within braces, at least one choice is required. Do not type the vertical bars on the command line.
Bold type	Bold type represents the name of an argument, an attribute, or a reason. It also represents the introduction of a new term.
<i>Italic type</i>	Italic type indicates important information, complete titles of manuals, or variables. Variables include information that varies in system output (Internal error <i>number</i>), in command lines (/PRODUCER= <i>name</i>), and in command parameters in text (where <i>dd</i> represents the predefined code for the device type).
UPPERCASE TYPE	Uppercase type indicates the name of a routine, the name of a file, or the abbreviation for a system privilege.
Monospace type	Monospace type indicates code examples and interactive screen displays. In the C programming language, monospace type in text identifies the following elements: keywords, the names of independently compiled external functions and files, syntax summaries, and references to variables or identifiers introduced in an example.
Bold monospace type	Bold monospace type indicates a command or command qualifier.
-	A hyphen at the end of a command format description, command line, or code line indicates that the command or statement continues on the following line.
Numbers	All numbers in text are assumed to be decimal unless otherwise noted. Nondecimal radixes—binary, octal, or hexadecimal—are explicitly indicated.

Chapter 1. MDMS DCL Commands

Warning

The tape functionality is not supported in ABS/MDMS V4.8. Disregard any mentions of tape throughout this document.

MDMS is the Media, Device, and Management Services xComponent of the Archive Backup System (ABS) and Hierarchical Storage Management (HSM) products.

You can use MDMS commands to manage all the databases for ABS. Alternatively, the pre-V4 ABS DCL commands are also available.

MDMS commands are also used to manipulate the media management database for HSM, whereas SMU manages the HSM database.

Important

VSI recommends using MDMS commands in preference to ABS commands whenever possible. MDMS provides a consistent and integrated interface for system management. ABS commands are retained for compatibility and may provide more limited functionality.

MDMS ALLOCATE DRIVE

MDMS ALLOCATE DRIVE — Selects and allocates a drive based on specifications provided. The selected drive name is assigned to the optional process logical name, which may be used in subsequent commands by that process.

Format

MDMS ALLOCATE DRIVE [*drive-name*]

Equivalent STORAGE Command: **STORAGE SELECT**

Parameters

drive-name

The name of a specific drive to allocate, which must already have been created with a **CREATE DRIVE** command. This parameter is optional: drives can also be allocated on the basis of volume, media type, location or node, and jukebox, or certain combinations thereof.

The maximum length of the drive name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

Description

The **MDMS ALLOCATE DRIVE** command selects and allocates a drive based on one of the following:

- A specified drive
- A specified volume

- A specified media type and optionally location, node, or group
- A specified jukebox

The selected local OpenVMS device name and the drive name are assigned to the optional logical name in a search list. Both MDMS and non-MDMS DCL commands may be issued using the logical name. The logical name remains assigned until it is deassigned, the allocating process terminates, or a **MDMS DEALLOCATE DRIVE** command specifying the logical name is issued.

The **MDMS ALLOCATE DRIVE** command allocates the selected drive to the current process. After the command is issued, the name of the selected drive is displayed on the terminal screen. The DCL **SHOW LOGICAL** command also displays the OpenVMS local device name and the drive name.

MDMS supports allocation of both local drives and remote drives (using the RDF software).

MDMS attempts to allocate a drive local to the node performing the allocation, if one matches the selection criteria. Failing that, a TMSCP-accessible drive is chosen next. If that fails, an RDF-accessible remote drive is chosen.

Privileges

The request requires MDMS_ALLOCATE_ALL or MDMS_ALLOCATE_OWN.

MDMS_ALLOCATE_OWN requires the specification of an owned volume for selection. All other selections including the drive name parameter require MDMS_ALLOCATE_ALL.

MDMS_ASSIST is also required unless the **/NOASSIST** qualifier is specified.

Restrictions

The drive-name parameter cannot be used with the **/MEDIA_TYPE**, **/JUKEBOX**, **/LOCATION**, **/NODE**, **/GROUP**, or **/VOLUME** qualifiers.

The **/JUKEBOX** qualifier cannot be used with **/GROUP**, **/LOCATION**, **/NODE**, **/VOLUME**, or the *drive-name* parameter.

The **/MEDIA_TYPE** qualifier cannot be used with **/VOLUME** or the *drive-name* parameter.

The **/VOLUME** qualifier cannot be used with **/GROUP**, **/LOCATION**, **/MEDIA_TYPE**, **/NODE**, **/JUKEBOX** or the *drive-name* parameter.

The **/GROUP**, **/JUKEBOX**, **/LOCATION**, and **/NODE** qualifiers are mutually exclusive. If one is specified, then **/MEDIA_TYPE** must also be specified (except for **/JUKEBOX**).

The **/[NO]PREFERRED** qualifier is ignored if **/VOLUME** is not supplied.

Qualifiers

/ASSIST (Default)

/NOASSIST

The default **/ASSIST** qualifier outputs an operator request to allocate a drive if there is a problem allocating a drive immediately. If the operator responds positively to the operator message, the request is retried, otherwise the request fails. The **/NOASSIST** qualifier performs the operation

without operator assistance, and uses the **/RETRY** and **/INTERVAL** qualifiers to perform retries. If all retries are exhausted the command fails.

The right MDMS_ASSIST is required unless **/NOASSIST** is specified.

/DEFINE=logical-name

Specifies the logical name to be assigned for the drive. The logical name is a process logical name.

The maximum length of the logical name is 31 characters.

The logical name is assigned to an equivalence string containing the drive name and allocated VMS device name in a search list; as such, both MDMS and non-MDMS commands (e.g. **MOUNT**) can be issued on the logical name.

/GROUP=group-name

When used with **/MEDIA_TYPE**, you can specify the name of the group (of nodes supporting the drive) from which to select and allocate the drive. Only one group may be specified.

This qualifier requires the right MDMS_ALLOCATE_ALL.

/INTERVAL=delta-time

Specifies the interval between retries when no drives are available. If not specified, the default interval is one minute.

/JUKEBOX=jukebox-name

The name of the jukebox from which the drive will be selected.

The maximum length of the jukebox name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

If **/MEDIA_TYPE** is also specified, the drive must support the specified media type as well as being in the jukebox.

This qualifier requires the right MDMS_ALLOCATE_ALL.

/LOCATION=location

When used with **/MEDIA_TYPE**, this qualifier specifies the location of the node from which to select and allocate the drive. Only one location may be specified.

This qualifier requires the right MDMS_ALLOCATE_ALL.

/MEDIA_TYPE=media-type

The distinct media type that the drive must support. This media type must match one defined for a drive in the media type attribute (for read/write allocation), or in the read only media type attribute (for read-only allocation).

This qualifier requires the right MDMS_ALLOCATE_ALL.

/NODE=node-name

When used with **/MEDIA_TYPE**, you can specify the name of the node from which to select and allocate the drive. Only one node may be specified.

This qualifier requires the right MDMS_ALLOCATE_ALL.

/PREFERRED**/NOPREFERRED (Default)**

The **/PREFERRED** qualifier tries to allocate the preferred drive for a volume when **/VOLUME** is entered; this is the last drive that the volume was loaded in, if that drive is available. The default **/NOPREFERRED** forces a round-robin drive selection.

/REPLY=symbol

The name of the symbol to receive the operator's reply when operator intervention is needed. The symbol will contain the operator reply to the DCL **REPLY/TO** or **REPLY/ABORT** command. The maximum length of a symbol name is 31 characters.

This qualifier is only applicable when the **/ASSIST** qualifier is specified.

/RETRY_LIMIT=number**/NORETRY_LIMIT**

Specifies if MDMS should retry the operation if no drive is available and **/ASSIST** is not specified. **/NORETRY_LIMIT** means that there is no limit on the number of retries. Retries are performed at the delta time specified by **/INTERVAL**. If all retries are exhausted, the operation fails.

The default is **/RETRY_LIMIT=0**, which means that no retries are performed.

/VOLUME=volume-id

The volume ID of the volume for which an appropriate drive will be selected that can support the volume's media type and placement. If a volume is specified that currently resides in a jukebox, the drive selected will be from the same jukebox if the drives in the jukebox support the media type of the volume.

The maximum length of the volume ID is 6 characters. Valid characters are " , ! , " , % - ? , A - Z , _ " .

/WRITE (Default)**/NOWRITE**

Specifies that the drive is to be allocated for read-only operations only. This potentially makes a larger pool of drives available for allocation for certain media types.

The default is WRITE.

Examples

The following command allocates the drive \$1\$MUA5 and assigns the drive name to the MYDRIVE logical name. Note that the drive name is the same as the VMS device name in this case:

```
$ MDMS ALLOCATE DRIVE $1$MUA5 /DEFINE=MYDRIVE
```

The following command allocates drive JUKE_2_DRIVE_1, requests operator assistance if the allocation fails, and stores the operator's reply message in symbol ALLOC_REPLY:

```
$ MDMS ALLOCATE DRIVE JUKE_2_DRIVE_1 /ASSIST -
```

```
__$ /REPLY=ALLOC_REPLY
```

The following command allocates a drive that supports volume LAB003 for read-only operations, and requests a round-robin drive selection rather than use the preferred drive:

```
$ MDMS ALLOCATE DRIVE /VOLUME=LAB003 /NOWRITE -  
__$ /NOPREFERRED
```

The following command allocates one of the drives in jukebox TESTJUKE and assigns the drive name to the MYDRIVE logical name:

```
$ MDMS ALLOCATE DRIVE /JUKEBOX=TESTJUKE -  
__$ /DEFINE=MYDRIVE
```

The following command allocates a drive that supports the TK85K media type on node CRUMBS, and specifies a retry interval of 10 seconds if a drive is not immediately available with no limit on retries:

```
$ MDMS ALLOCATE DRIVE /MEDIA_TYPE=TK85K -  
__$ /NODE=CRUMBS /INTERVAL=00:00:10 /NORETRY_LIMIT
```

MDMS ALLOCATE VOLUME

MDMS ALLOCATE VOLUME — Allocates available free volumes to the requesting user or a specified user.

Format

```
MDMS ALLOCATE VOLUME [volume-id]
```

Equivalent STORAGE Command: **STORAGE ALLOCATE**

Parameters

volume-id

Specifies the volume to allocate. This parameter is optional, and volumes can be allocated on the basis of **/BIND**, **/JUKEBOX**, **/LIKE_VOLUME**, **/LOCATION**, **/MEDIA_TYPE** or **/POOL** instead.

The maximum length of the volume ID is 6 characters. Valid characters are " ,! , % - ? , A - Z , _ " .

Description

The **MDMS ALLOCATE VOLUME** command allocates available free volumes to the requesting user or a specified user. The newly allocated volumes can also be appended to the end of a volume set. A volume set is defined to be one or more allocated volumes.

If the *volume-id* parameter is used, that specific volume will be allocated if it is in the free state. If the *volume-id* parameter is not used, a free volume will be selected based on one or more of the following selection criteria:

- Bind volume

- Jukebox
- Like volume
- Location
- Pool
- Media type

Privileges

The request requires MDMS_ALLOCATE_ALL or MDMS_ALLOCATE_POOL.

MDMS_ALLOCATE_POOL requires the volume to be allocated from a named pool to which the calling user is authorized. All other allocations, including allocation from the scratch (unnamed) pool and the specification of **/USER_NAME**, require MDMS_ALLOCATE_ALL.

The **/BIND** qualifier requires MDMS_BIND_OWN, or MDMS_BIND_ALL if the **/USER_NAME** qualifier is specified.

The **/BLOCK_FACTOR**, **/DESCRIPTION**, **/FORMAT**, **/RECLENGTH**, **/[NO] SCRATCH_DATE**, and **/[NO] TRANSITION_TIME** qualifiers require MDMS_SET_OWN or MDMS_SET_POOL. If the **/USER_NAME** qualifier is specified, then they require MDMS_SET_ALL or MDMS_SET_VOLUME.

MDMS_ASSIST is also required unless the **/NOASSIST** qualifier is specified.

Restrictions

The *volume-id* parameter cannot be used with the **/JUKEBOX**, **/LIKE_VOLUME**, **/LOCATION**, **/POOL**, or **/QUANTITY** qualifiers.

/BIND cannot be used with the **/LIKE_VOLUME**, **/JUKEBOX**, **/LOCATION**, **/MEDIA_TYPE**, or **/POOL** qualifiers.

/LIKE_VOLUME cannot be used with the **/BIND**, **/JUKEBOX**, **/LOCATION**, **/MEDIA_TYPE**, or **/POOL** qualifiers.

/MEDIA_TYPE is required when **/JUKEBOX**, **/LOCATION**, or **/POOL** are specified.

/MEDIA_TYPE is also required when allocating a specific volume that has multiple media types.

Qualifiers

/ASSIST (Default)

/NOASSIST

The default **/ASSIST** qualifier outputs an operator request to allocate a volume if there is a problem allocating a volume. If the operator responds positively to the operator message, the request is retried, otherwise the request fails. The **/NOASSIST** qualifier performs the operation without operator assistance, and fails if there is a problem allocating a volume.

The right MDMS_ASSIST is required unless **/NOASSIST** is specified.

/BIND=volume-id

The new allocated volume or volume set is appended to the volume set specified by the *volume-id*. The *volume-id* specifies the selection criteria for allocating the new volumes.

The maximum length of the volume ID is 6 characters. Valid characters are " , ! , " , % - ? , A - Z , _ " .

When the **/BIND** qualifier is used, the following attributes must match:

- Media type
- Pool
- Placement (in same jukebox, magazine, or location)

The BIND volume must be allocated.

This qualifier requires the right MDMS_BIND_OWN, or MDMS_BIND_ALL if the **/USER_NAME** qualifier is used.

/BLOCK_FACTOR=number

Modifies the block factor attribute of the allocated volume records. If not specified, the block factor attribute is not changed.

This qualifier requires the right MDMS_SET_OWN or MDMS_SET_POOL. It requires MDMS_SET_ALL or MDMS_SET_VOLUME if the **/USER_NAME** qualifier is used.

/DEFINE=logical-name

Specifies a logical name to be assigned for the volume allocated. The logical name is a process logical name. If the **/QUANTITY** qualifier is used, the returned volume will be the first volume in the volume set. The maximum length of the logical name is 31 characters.

/DESCRIPTION="text"

Modifies comments about the object in the volume record. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters.

To clear the existing description, specify "". If not specified, the volume description is not changed.

This qualifier requires the right MDMS_SET_OWN or MDMS_SET_POOL. It requires MDMS_SET_ALL or MDMS_SET_VOLUME if the **/USER_NAME** qualifier is used.

/FORMAT=keyword

This qualifier modifies the format field in the volume record. If not specified, the format attribute is not changed. Valid values are:

- ASCII
- BACKUP
- EBCDIC

- NONE
- RMUBACKUP

This qualifier requires the right MDMS_SET_OWN or MDMS_SET_POOL. It requires MDMS_SET_ALL or MDMS_SET_VOLUME if the **/USER_NAME** qualifier is used.

/JUKEBOX=jukebox-name

When specified with **/MEDIA_TYPE**, allocates a volume that resides in the specified jukebox.

/LIKE_VOLUME=volume-id

Directs that the allocated volume has attribute values identical to those specified to the *volume-id*. The following attributes define a "like volume":

- Media type
- Pool
- Placement (in the same jukebox, magazine, or location)

/LOCATION=location

When specified with **/MEDIA_TYPE**, specifies the location from which to allocate volumes. If a volumes placement is OFFSITE then the off-site location is used. Otherwise the volume's ONSITE location is used. If not specified, other selection criteria are used to allocate the volumes.

/MEDIA_TYPE=media-type

Specifies that the volumes are allocated with the specified media type. If not specified, other selection criteria are used to allocate the volumes. This qualifier is required when allocating a specific volume that currently has multiple media types defined for it.

/POOL=pool-name

The pool from which the volumes will be selected.

The maximum length of the pool name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

The user must be authorized for the pool unless they have MDMS_ALLOCATE_ALL.

/QUANTITY=number

The number of volumes to allocate. Use this qualifier to allocate more than one volume. The default value is 1. If you enter a number greater than 1, the allocated volumes are bound together in one volume set. If you use the **/BIND** qualifier, these allocated volumes are bound to the end of the volume (set) specified in the **/BIND** qualifier.

/RECLength=number

Modifies the volume's record length attribute. If not specified, the record length attribute is not changed.

This qualifier requires the right MDMS_SET_OWN or MDMS_SET_POOL. It requires MDMS_SET_ALL or MDMS_SET_VOLUME if the **/USER_NAME** qualifier is used.

/REPLY=symbol

The name of the symbol to receive the operator's reply when operator intervention is needed. The symbol will contain the operator reply to the DCL **REPLY/TO** or **REPLY/ABORT** command. The maximum length of a symbol name is 31 characters.

This qualifier is only applicable when the **/ASSIST** qualifier is specified.

/SCRATCH_DATE=date***/NOSCRATCH_DATE***

Modifies the scratch date in the volume record. The scratch date is the date that the volume is placed in the TRANSITION state (or FREE state if the volume has no transition time). If **/NOSCRATCH_DATE** is specified, the volume will never be automatically deallocated. If not specified, the volume's scratch date is not changed.

This qualifier requires the right MDMS_SET_OWN or MDMS_SET_POOL. It requires MDMS_SET_ALL or MDMS_SET_VOLUME if the **/USER_NAME** qualifier is used.

/TRANSITION_TIME=delta-time***/NOTRANSITION_TIME***

Modifies the amount of time, as a delta time, that will be applied to the current scratch date to form the length of time the volumes will remain in the TRANSITION state before going into the FREE state.

Use the standard OpenVMS delta time format to specify a delta time for the duration.

If **/NOTRANSITION_TIME** is specified, the volume goes directly into the FREE state on deallocation. If not specified, the transition time in the volume record is not changed.

This qualifier requires the right MDMS_SET_OWN or MDMS_SET_POOL. It requires MDMS_SET_ALL or MDMS_SET_VOLUME if the **/USER_NAME** qualifier is used.

/USER_NAME=username

The user for whom the volume is being allocated. The username must exist on the system where the command is entered.

The maximum length of the username is 31 characters.

This qualifier requires the right MDMS_ALLOCATE_ALL.

Examples

The following command allocates three TK85K volumes to the current user. The volumes are bound in a new volume set:

```
$ MDMS ALLOCATE VOLUME /QUANTITY=3 /MEDIA_TYPE=TK85K
```

The following command allocates one TK85K volume at location CXO to the current user:

```
$ MDMS ALLOCATE VOLUME /MEDIA=TK85K /LOCATION=CXO
```

The following command allocates one volume with the same attributes as TEST01 to the current user. The new volume is bound to the end of the volume set containing TEST01:

```
$ MDMS ALLOCATE VOLUME /BIND=TEST01
```

The following command allocates volume USER30 to the current user and modifies the volume's description to "MAY REPORTS":

```
$ MDMS ALLOCATE VOLUME USER30 /DESCRIPTION="MAY REPORTS"
```

The following command allocates a volume with similar attributes to volume AGW500, for user SYSTEM:

```
$ MDMS ALLOCATE VOLUME /LIKE_VOLUME=AGW500 -  
/USER_NAME=SYSTEM
```

MDMS BIND VOLUME

MDMS BIND VOLUME — Binds a volume to another volume or volume set, or binds a volume set to another volume set.

Format

```
MDMS BIND VOLUME [volume-id]
```

Equivalent STORAGE Command: **STORAGE APPEND**

Parameters

volume-id

Specifies the volume ID of the volume to bind. If an existing volume set is to be bound to another volume or volume set, then specify the volume ID of the first member of the volume set to bind. If the *volume-id* parameter is omitted, a volume with similar characteristics to those in the set will be allocated and bound to the end of the volume set.

The maximum length of a volume ID is 6 characters. Valid characters are " ,! ,% - ? ,A-Z ,_ ".

Description

The **MDMS BIND VOLUME** command binds a volume to another volume or volume set or binds a volume set to another volume set. When binding to a volume or volume set, all involved volumes must be allocated and of the same media type.

When binding a volume set to another volume or volume set, the *volume-id* parameter must be the first member of the volume set. To append to the end of a volume set, use the **/TO_SET** qualifier.

The volumes in the volume set must already be allocated to the same user (username and UIC). When a new volume is bound to a volume set, it acquires the scratch date of the volume set.

Privileges

The request requires MDMS_BIND_ALL or MDMS_BIND_OWN.

If the **/USER_NAME** qualifier is not specified, MDMS_BIND_OWN allows the user to bind volumes that are allocated to them. Binding on behalf of another user with the **/USER_NAME** qualifier requires MDMS_BIND_ALL.

Restrictions

None.

Qualifiers

/TO_SET=volume-id

The volume ID of a volume or volume set member. The volume (set) specified in the *volume-id* parameter is appended to the end of the volume set containing this volume.

The maximum length of the volume ID is 6 characters. Valid characters are " ,! , % - ? , A - Z , _ " .

If the *volume-id* command parameter is not specified, then a volume similar to the volume in the **/TO_SET** qualifier is allocated and bound to the end of that volume set.

/USER_NAME=username

The user for whom the volume is being bound. The username must exist on the system where the command is entered. The maximum length of the username is 31 characters.

This qualifier requires the right MDMS_BIND_ALL.

Examples

Volume set one contains VOL001, VOL002, and VOL003. Volume set two contains VOL004, VOL005, and VOL006. The following command binds the volume set containing VOL001 to the volume set containing VOL006:

```
$ MDMS BIND VOLUME VOL001/TO_SET=VOL006
```

The created volume set will contain volumes VOL004, VOL005, VOL006, VOL001, VOL002, and VOL003. Volumes in both volume sets must have compatible attributes.

Volume VOL002 is a single volume and VOL005 is part of a volume set that contains VOL004, VOL005 and VOL006. The following command binds the volume VOL002 to the volume set containing VOL005:

```
$ MDMS BIND VOLUME VOL002 /TO_SET=VOL005 -  
/USER_NAME=SYSTEM
```

The created volume set will contain volumes VOL004, VOL005, VOL006, and VOL002. All the volumes involved are allocated to user SYSTEM.

MDMS CANCEL REQUEST

MDMS CANCEL REQUEST — Cancels a previously issued request. The request may have been issued either synchronously or asynchronously.

Format

```
MDMS CANCEL REQUEST [request-id] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

request-id

Specifies the request IDs of the requests to cancel. If the request ID is not known, the user can issue a **SHOW REQUESTS** command, which displays the request ID's of all outstanding requests.

Description

The **MDMS CANCEL REQUEST** command cancels the specified outstanding request. The following types of request can be cancelled using this command:

- ALLOCATE DRIVE
- ALLOCATE VOLUME
- INITIALIZE VOLUME
- INVENTORY JUKEBOX
- LOAD DRIVE
- LOAD VOLUME
- MOVE MAGAZINE
- MOVE VOLUME
- UNLOAD DRIVE
- UNLOAD VOLUME
- Any operation involving OPCOM

Privileges

The request requires MDMS_CANCEL_ALL, MDMS_CANCEL_OWN, or MDMS_CANCEL_POOL.

If you are canceling your own request, MDMS_CANCEL_OWN or MDMS_CANCEL_POOL is required. If you are canceling any other request, MDMS_CANCEL_ALL is required.

Restrictions

None.

Qualifiers

None.

Examples

The following command cancels the request with ID 812:

\$ MDMS CANCEL REQUEST 812

MDMS CREATE ARCHIVE

MDMS CREATE ARCHIVE — Creates a new archive definition in the MDMS database. Archives were formerly known as storage classes, archive classes, and storage policies.

Format

MDMS CREATE ARCHIVE [*archive-name*] [, ...]

Equivalent ABS Command: **CREATE STORAGE_CLASS**

Parameters

archive-name

Specifies the name of the archive.

The maximum length of the archive name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of archive names may be entered.

Description

The **MDMS CREATE ARCHIVE** command creates a new archive definition in the MDMS database. An archive describes where data is saved and its catalog information.

When creating a new object record, MDMS supplies default values on attributes you do not specify. Alternatively, they can be inherited from a specified archive using the **/INHERIT** qualifier.

Privileges

The request requires MDMS_CREATE_ALL.

The **/VOLUME_SETS** qualifier also requires MDMS_SET_PROTECTED, because this attribute is normally managed by MDMS. You should not modify this attribute unless you are trying to recover from an abnormal situation.

Restrictions

This command is restricted to nodes with an ABS license.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The **/EXPIRATION_DATE** and **/RETENTION_DAYS** qualifiers are mutually exclusive.

The following combinations of qualifiers are not allowed based on the archive type:

DISK:	/CONSOLIDATION	TAPE:	/DESTINATION

/DRIVES
/LOCATION
/MEDIA_TYPE
/POOL
/VOLUME_SETS

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all, or specific users on particular nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier, and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the *node*, *group*, and *username* portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/ARCHIVE_TYPE=*keyword*

The archive type specifies the type of archive media associated with the archive. The options are:

DISK	Files are saved to or restored from a disk. The destination may be filled in to specify the disk and directory where the archive data resides.

TAPE	Files are saved to or restored from a tape volume and MDMS is used to manage the tape volume.

The default is TAPE.

/CATALOG=(NAME=*catalog-name*, [NO]NODES=(*node*[,...]))

The **/CATALOG** qualifier specifies the name of the catalog used by the archive and a list of nodes and/or groups that can directly access the catalog. Specify a local catalog name representing a catalog object in the MDMS database. With the NODES keyword, you may specify nodes, groups, or a combination thereof. If you specify NONODES, the local execution node on which a save or restore is performed is used as the catalog node.

By default, the catalog name "ABS_CATALOG" is used. For an archive named "DISASTER_RECOVERY", the default is no catalog name.

The default for nodes is NONODES.

/CONSOLIDATION=(*option*[,...])

The **/CONSOLIDATION** qualifier specifies the criteria under which new volume sets are created for this archive. You may specify one or more of the following options:

- **INTERVAL=*delta-time*** – The time to elapse from the creation of the last volume set to the creation of a new volume set. To specify an unlimited consolidation interval, specify NOINTERVAL.
- **SAVESETS=*number*** – The maximum number of savesets to exist on a volume set before a new volume set is created. Zero means no limit.
- **VOLUMES=*number*** – The maximum number of volumes which may be added to the current volume set. Zero means no limit.

A new volume set is created when any one of the consolidation criteria expire. By default, the consolidation criteria is set to the following:

```
INTERVAL=7-00:00:00, SAVESETS=0, VOLUMES=0
```

If no consolidation criteria are specified, new volume sets are not created unless an error occurs. This qualifier is allowed only with archive type TAPE.

/DESCRIPTION="*text*"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/DESTINATION=*disk*:[*dir*]

If the archive type is DISK, the **/DESTINATION** qualifier specifies the disk and directory where the archive data resides. If no destination is specified for archive type DISK, the following is used by default:

```
ABS$ROOT:[000000]
```

This qualifier is allowed only with archive type DISK.

/DRIVES=(*drive-name*[,...])
/NODRIVES (Default)

The **/DRIVES** qualifier specifies a list of tape drives to be used by the archive. Specify one or more drive names. By default, no drives are specified, meaning that any drive satisfying other compatibility criteria may be used.

This qualifier is allowed only with archive type TAPE.

/EXPIRATION_DATE=*absolute-time*
/NOEXPIRATION_DATE

The **/EXPIRATION_DATE** qualifier specifies the date that saved data will expire. Specify an absolute or delta date/time, or a reserved word such as TODAY or TOMORROW. Alternatively, use the relative **/RETENTION_DAYS** qualifier to specify the number of days the saved data should be retained. **/NOEXPIRATION_DATE** means that the data is retained indefinitely.

By default, data is retained for 365 days.

/INHERIT=*archive-name*

This qualifier allows you to specify an existing archive record from which the new object record inherits attribute values. MDMS supplies default values if you specify none. All attributes may be inherited with the exception of Archive Name.

/LOCATION=*location-name*
/NOLOCATION

The **/LOCATION** qualifier specifies the location to be used to allocate a drive to load a scratch volume (for example, **HELP MDMS ALLOCATE DRIVE/LOCATION**). Specify a location name specified in the database. This qualifier is applicable only for archive type TAPE. **/NOLOCATION** removes location as a selection criterion for volume allocation.

By default, the domain default on-site location is used.

/MAXIMUM_SAVES=*number*

The **/MAXIMUM_SAVES** qualifier specifies the maximum number of save operations that can execute simultaneously using this archive. For example, you can create three save requests and schedule them to run simultaneously, assuming that there are enough resources (e.g. drives) available. Valid values are 1–36. The default is 1.

/MEDIA_TYPE=*media-type*

The **/MEDIA_TYPE** qualifier specifies the media type of tape volumes used for this archive. By default, the domain default media type is used if the archive type is TAPE, otherwise none.

This qualifier is allowed only with archive type TAPE.

/OWNER_NAME=*node::username*
/OWNER_NAME=*group::username*
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/POOL=pool-name

/NOPOOL (Default)

The **/POOL** qualifier specifies the pool from which volumes will be allocated. Specify a pool name that exists in the database. If you specify **/NOPOOL**, then only volumes with no pool definition may be used (equivalent to the SCRATCH pool).

By default, no pool is defined.

This qualifier is allowed only with archive type TAPE.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/RETENTION_DAYS=number

The **/RETENTION_DAYS** qualifier specifies the number of days to retain data for this archive. Either this qualifier or **/EXPIRATION_DATE** may be used to specify this information. If neither an expiration date nor retention days are specified, then the data is retained indefinitely.

By default, a retention of 365 days is used.

/VOLUME_SETS=(volume-id[,volume-id])

/NOVOLUME_SETS

The **/VOLUME_SETS** qualifier specifies the active volume sets being used (for write) by the archive. Each *volume-id* should represent the first volume in the volume set.

/NOVOLUME_SETS clears the volume sets, and forces a new volume set to be allocated when the archive is next used.

Setting of volume sets is not recommended as ABS normally assigns volume sets to archives. This is a protected field.

This qualifier requires the right MDMS_SET_PROTECTED.

This qualifier is allowed only with archive type TAPE.

Examples

The following command creates an archive `USER_BACKUPS`, specifying access control for user `SMITH` on all nodes and user `JONES` on node `NOD001` to have all allowed accesses. It also specifies a consolidation criterion of 10 volumes per volume set and a retention period of 30 days:

```
$ MDMS CREATE ARCHIVE USER_BACKUPS -  
_ $ /ACCESS_CONTROL=(USER=*::SMITH, -  
_ $ ALLOW=(SET,SHOW,READ,WRITE,EXECUTE,DELETE,CONTROL)) -  
_ $ /CONSOLIDATION=VOLUMES=10 -  
_ $ /RETENTION_DAYS=30
```

MDMS CREATE CATALOG

MDMS CREATE CATALOG — Creates a new catalog definition in the local ABS database.

Format

```
MDMS CREATE CATALOG [catalog-name] [, ...]
```

Equivalent ABS Command: **RUN SYS\$SYSTEM:ABS\$CATALOG_OBJECT**

Parameters

catalog-name

Specifies the name of the catalog.

The maximum length of the catalog name is 26 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

Description

The **MDMS CREATE CATALOG** command creates a new catalog definition in the local ABS database. A catalog describes how, where and what information about saved data is recorded.

When creating a new object record, MDMS supplies default values on attributes you do not specify.

Privileges

The request requires **MDMS_CREATE_ALL**.

Restrictions

This command is restricted to nodes with an ABS license.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier, and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

Only ONE access control entry is allowed for a catalog object.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/DIRECTORY=*disk*:[*dir*]

The **/DIRECTORY** qualifier specifies the disk and directory specification where the catalog files are created.

If no directory is specified, the default is ABS\$CATALOG.

If the catalog files already exist no newer versions are created.

The directory must be included in the search list for the ABS\$CATALOG logical name.

/NODE=*node-name*

The **/NODE** qualifier specifies on which node the new catalog will be created.

The default is the node on which the **CREATE CATALOG** command is entered.

/OWNER_NAME=node::username
/OWNER_NAME=group::username
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/STAGING (Default)

/NOSTAGING

The **/STAGING** qualifier turns on the use of a staging catalog during saves. Using a staging catalog can greatly speed up a save operation by deferring update of the full catalog to an off-line operation. However, a restore cannot be performed until the full catalog is updated. Use of staging is highly recommended.

The **/NOSTAGING** qualifier turns off use of a staging catalog.

Staging is only used on catalog types **FILES** and **VOLUME_SETS**.

By default, staging is enabled.

/TYPE=keyword

The **/TYPE** qualifier defines the type of catalog being used. Select one of the following options:

DISKS	This type of catalog records only information about disk volumes. As such, restores must apply to whole disks; individual file restores are not possible from this type of catalog. However, this type of catalog is much smaller than the other types.
FILES (Default)	This type of catalog records both disk and file information, and there is one set of catalog files for the catalog object. This option produces a low number of files to maintain but recovering disk space from deleted entries requires

	manual intervention on the catalog. This type was used in previous versions of ABS and is available for compatibility purposes.
ORACLE_DB	This type of catalog records the backup file names for a save of a database using the System Backup to Tape interface.
VOLUME_SETS	This type of catalog records both disk and file information, and there is one set of catalog files for each volume set. This option produces a much larger number of catalog files, but the catalog cleanup and associated deleted space recovery is optimal. Catalogs space is reclaimed by simply deleting the catalog file when the associated volume set reaches its scratch date.
SLS	This type of catalog is used only for the lookup of the files backed up by SLS.

The default and recommended value is FILES.

Examples

The following command creates a catalog JOE_CATALOG, specifying user JOE on node MYJOE with selected allowances, with the catalog residing on MYJOE : :DISK\$USER : [JOE] and staging enabled:

```
$ MDMS CREATE CATALOG JOE_CATALOG -
_$ /ACCESS_CONTROL=(USER=MYJOE::JOE, -
_$ALLOW=(SET, SHOW, EXECUTE, DELETE, READ, WRITE)) -
_$ /DIRECTORY=DISK$USER:[JOE]
_$ /NODE=MYJOE
_$ /STAGING
```

MDMS CREATE DRIVE

MDMS CREATE DRIVE — Creates a new drive definition in the MDMS database.

Format

```
MDMS CREATE DRIVE [drive-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

drive-name

Specifies the name of the drive. The OpenVMS device name must also be specified with the **/DEVICE** qualifier if it is different from the drive name. The drive name must be unique in the MDMS domain. You should not specify a node in the drive name; rather, you should specify either a list of nodes or groups that have access to the drive in the **/NODES** or **/GROUPS** qualifiers.

The maximum length of a drive name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of drive names may be entered.

Description

The **MDMS CREATE DRIVE** command creates a new drive definition in the MDMS database.

When creating a new object record, MDMS supplies default values on attributes you do not specify. Alternatively, they can be inherited from a specified drive using the **/INHERIT** qualifier.

Privileges

The request requires MDMS_CREATE_ALL.

The **/STATE** qualifier also requires MDMS_SET_PROTECTED.

Since this attribute is normally managed by MDMS. You should not modify this attribute unless you are trying to recover from an abnormal situation.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The **/JUKEBOX** and **/STACKER** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS=keyword

This qualifier defines the type of access to the drive, which can be one of the following keywords:

ALL (Default)	Supports local node/cluster/fibre access and remote (RDF) access
LOCAL	Supports local node/cluster/fibre access only
REMOTE	Supports remote (RDF) access only

Access to drives is restricted on allocate requests; for example, it is not possible to allocate a drive designated as local access remotely using RDF. However, with the proper rights it is possible to issue other MDMS commands (such as **LOAD**) both locally and remotely.

The default is ALL.

/ACCESS_CONTROL=(USERNAME=user-id, ALLOW=(option[,...])) **/NOACCESS_CONTROL (Default)**

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object

- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/AUTOMATIC_REPLY (Default)

/NOAUTOMATIC_REPLY

Specifies that MDMS automatically replies to all OPCOM messages that can be polled for completion on requests for this particular drive.

The default is **/AUTOMATIC_REPLY**.

/DESCRIPTION="text"

Comments about the drive. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/DEVICE=*vms-device-name*

Specifies the OpenVMS device name of the drive. This qualifier is required if the OpenVMS device name is different from the drive name. Do not specify a node specification (*NODE::name*) in the *vms-device-name* parameter. Instead use the **/NODES** or **/GROUPS** qualifier.

The default device name is the drive name.

/DISABLED

Places the drive in the disabled state. This prevents the drive from being selected and allocated for use. This takes effect immediately. However, if the drive is already in use, operations on that drive will continue until the drive is deallocated.

The default is **/ENABLED**.

/DRIVE_NUMBER=*number*

This qualifier defines the drive number for robot commands if the drive is in a jukebox. This qualifier must be specified for multi-drive, MRD-controlled jukeboxes.

The default is drive number 0.

/ENABLED (Default)

Places the drive in the enabled state. This allows the drive to be selected and allocated for use. This takes effect immediately.

The default is **/ENABLED**.

/GROUPS=(group-name[,...])
/NOGROUPS

Specifies the names of groups of nodes that share common access to this device. Usually, only one group is specified. If neither the **/NODES** nor **/GROUPS** qualifier is specified, then the node from which the command was issued is used as the node name. Groups can also be specified with the **/NODES** qualifier.

/INHERIT=drive-name

This qualifier allows you to specify an existing drive record from which the new object record inherits attribute values. MDMS supplies default values if you specify none. All attributes may be inherited with the exception of the following: Drive Name, Device Name, State.

/JUKEBOX=jukebox-name

If the drive is in a jukebox, this qualifier specifies the jukebox name.

/MEDIA_TYPE=(media-type[,...])
/NOMEDIA_TYPE

Specifies one or more distinct media types that the drive can support for read-write access. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace objects in the list, rather than adding them by default. The **/NOMEDIA_TYPES** qualifier removes all media types. When a drive is created with no media types, the default media type from the domain is used.

/NODES=(node-name[,...])
/NONODES

Specifies one or more distinct nodes that have direct access to the drive. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace objects in the list, rather than adding them by default. The **/NONODE** qualifier removes all nodes. If neither the **/NODES** nor **/GROUPS** qualifier is specified, then the node from which the command was issued is used as the node name.

/OWNER_NAME=node::username
/OWNER_NAME=group::username
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/READONLY=(media-type[,...])
/NOREADONLY

Specifies one or more distinct media types that the drive can support for read-only access. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace objects in the list, rather than adding them by default. The **/NOREADONLY** qualifier removes all read-only media types.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/SHARED
/NOSHARED (Default)

The **/SHARED** qualifier defines that the drive may be used by non-MDMS clients and that the drive is only partially managed. If the drive is set to the default **/NOSHARED**, the MDMS server allocates the drive at all times it is not used by an application or user. Setting the drive to **/SHARED** clears this allocation.

The default is **/NOSHARED**.

/STACKER
/NOSTACKER (Default)

The **/STACKER** qualifier indicates that the drive is to be treated as a stacker gravity loader. The default **/NOSTACKER** indicates that the drive is to be treated as a standalone drive, or a robotically-controlled jukebox, as appropriate.

The default is **/NOSTACKER**.

/STATE=state

This is a protected field that should be modified only to recover on error. Use the **LOAD** and **UNLOAD** commands to manipulate the state field under normal operation. The **/STATE** qualifier sets the current drive state. The valid keywords are Empty, Full, Loading, and Unloading.

This qualifier requires the right MDMS_SET_PROTECTED.

Examples

The following command creates a local drive called \$1\$MUA5 with default parameters supplied by MDMS. In this example, the drive name is equivalent to the OpenVMS device name and the current node is used as the drive's node name:

```
$ MDMS CREATE DRIVE $1$MUA5:
```

The following command creates a local drive called `DRIVE_1` for OpenVMS device `$1$MUA5` on node `FARLFT`:

```
$ MDMS CREATE DRIVE DRIVE_1 /DEVICE=$1$MUA5:/NODE=FARLFT
```

The following command creates a drive with drive name `JUKE_1_DRIVE_1` and OpenVMS device name `$1$MKA500` in group `COOKIE`, for local and remote access, shared by non-MDMS users, and supporting media types `TK85K` and `TK88K`. The drive is part of jukebox `JUKE_1` with drive number 1:

```
$ MDMS CREATE DRIVE JUKE_1_DRIVE_1 -  
/MEDIA_TYPES=(TK85K, TK88K) /DEVICE=$1$MKA500 -  
/GROUPS=COOKIE /SHARE /JUKEBOX=JUKE_1/DRIVE_NUMBER=1 /ACCESS=ALL
```

The following command creates a drive object record named `FRED` for remote access. MDMS supplies all default attribute values:

```
$ MDMS CREATE DRIVE FRED /ACCESS=REMOTE
```

The following command defines drive `SPARKY_5`, device `$1$MUA5` on node `SPARKY`, that supports media type `TK85K`, supports both local and remote access, and is located on remote node `SPARKY`. The drive is not available for immediate use:

```
$ MDMS CREATE DRIVE SPARKY_5 /DEVICE=$1$MUA5 -  
/NODE=SPARKY /DISABLE /MEDIA_TYPES=TK85K /ACCESS=ALL
```

MDMS CREATE ENVIRONMENT

MDMS CREATE ENVIRONMENT — Creates a new environment definition in the MDMS database. Environments were formerly known as execution environments or environment policies. **ENVIRONMENT** is the new common name.

Format

```
MDMS CREATE ENVIRONMENT [environment-name] [, ...]
```

Equivalent ABS Command: **CREATE ENVIRONMENT**

Parameters

environment-name

Specifies the name of the environment.

The maximum length of the environment name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of environment names may be entered.

Description

The **MDMS CREATE ENVIRONMENT** command creates a new environment definition in the MDMS database. An environment describes how data is saved and restored during execution.

When creating a new object record, MDMS supplies default values on attributes you do not specify. Alternatively, they can be inherited from a specified environment using the **/INHERIT** qualifier.

Privileges

The request requires MDMS_CREATE_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ACTION=*keyword*

The **/ACTION** qualifier specifies an operation to perform on files or data being saved using this environment. Select one of the following options:

RECORD_DATE	Modifies the BACKUP date on files being saved (VMS_FILES only)

NO_CHANGE	Does not perform any operation on the data being saved
DELETE_OBJECT	Deletes the data being saved—this is typically used for long-term archive operations

If not specified, the default is RECORD_DATE.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list.

This is the default when specifying list items.

/ASSIST (Default)

/NOASSIST

The **/ASSIST** qualifier specifies whether saves and restores using this environment utilize operator assistance (using OPCOM) when needed to continue the operation.

If **/ASSIST** is specified and operator intervention is needed an OPCOM message will be output. If **/NOASSIST** is specified and an operator intervention is needed, the operation will fail and the next operation will be performed (if possible).

The default is **/ASSIST**.

/DATA_SAFETY=(keyword[,...])

/NODATA_SAFETY

The **/DATA_SAFETY** qualifier specifies one or more options that the backup agent uses to ensure the integrity of the data being saved. The qualifier is applicable to archive type TAPE and when using OpenVMS Backup as the backup agent. Select one or more of the following options:

CRC	Performs a Cyclic Redundancy check and writes it for each data block on a tape volume
FULL_VERIFY	Rereads all saved data and compares to what is on disk
XOR	If the CRC check detects a bad block during a restore operation, the XOR mechanism allows recovery of the block.

By default, data safety options are added. However, you can use the **/REMOVE** qualifier to remove any option, and the **/REPLACE** qualifier to replace the option list. If you specify **/NODATA_SAFETY**, then all options are removed.

If not specified, the default options are: (CRC, FULL_VERIFY, XOR).

/DESCRIPTION="text"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/DRIVE_COUNT=number

The **/DRIVE_COUNT** qualifier specifies the number of drives to use for each save or restore using this environment. If there are at least as many drives available as the drive count, that number

of drives are allocated for each save and restore request. If not, a reduced number of drives are allocated.

The default and highly recommended value is 1. The number of drives may range from 1 to 32.

/EPILOGUE="string"
/NOEPILOGUE (Default)

The **/EPILOGUE** qualifier specifies a string to execute after an entire save or restore operations completes. The string may be up to 80 characters and must be enclosed in quotes.

By default, no epilogue command is executed.

/INHERIT=environment-name

This qualifier allows you to specify an existing environment record from which the new object record inherits attribute values. MDMS supplies default values if you specify none. All attributes may be inherited with the exception of Environment Name.

/INTERVAL=delta-time

The **/INTERVAL** qualifier specifies the time between retries if a save or restore operation fails. This is used in association with the **/RETRY_LIMIT** qualifier. The interval range must be between 1–60 minutes.

If **/RETRY_LIMIT** is greater than zero, then the default interval applied is 00:15:00 (15 minutes).

/LISTING_OPTION=keyword

The **/LISTING_OPTION** qualifier specifies the level of detail in the list file for a save or restore operation. Select one of the following options:

NONE (Default)	No listing file is generated
BRIEF	Generates a brief listing file
FULL	Generates a full listing file including full file names and file information provided by the backup agent

If not specified, the default is NONE.

/LOCK
/NOLOCK (Default)

The **/LOCK** qualifier specifies the action to take when data usage conflicts occur. If **/LOCK** is specified, data is saved even if other applications have the data locked for write access. If **/NOLOCK** is specified, the data is not saved if other applications have the data locked for write access.

The default is **/NOLOCK**.

/NOTIFICATION=(option[,...])
/NONOTIFICATION

The **/NOTIFICATION** qualifier specifies options to use for notifying mail user or OPCOM classes when using this environment. Select one or more of the following options:

- **MAIL**=(*username*[,...]) – Specifies one or more mail users to receive mail on completion of an operation. The username may include a node name (*node::username*) or group name (*group::username*). By default the mail is sent to the user on the node on which the request executes.
- **OPCOM**=(*class*[,...]) – Specifies one or more OPCOM classes to receive an OPCOM on completion of an operation. The class names must be valid OpenVMS OPCOM class names (TAPES etc).
- **TYPE**=*keyword* – Specifies the level of detail included in the notification. This keyword is applicable to MAIL users only. Select one of the following options:
 - BRIEF
 - FULL
 - NORMAL
- **WHEN**=(*option*[,...]) – Specifies when the notification is sent and under what circumstances. Select one or more of the following options:
 - COMPLETE – Sends notification at the completion of a request with any status
 - ERROR – Sends notification at the completion of a request with error or fatal status
 - FATAL – Sends notification at the completion of a request with fatal status
 - START – Sends notification at the start of a request
 - WARNING – Sends notification at the completion of a request with warning, error or fatal status

Only one notification entry can be entered per command. Use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to add, remove or replace mail users, OPCOM classes or when definitions. The usernames in an entry must match exactly to successfully modify or remove an entry.

If not specified, the default is:

(OPCOM=TAPES, TYPE=BRIEF, WHEN=FATAL)

/OWNER_NAME=*node::username*
/OWNER_NAME=*group::username*
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/PROFILE=(USER=*username*, *option*[,...])

The **/PROFILE** specifies the execution user context when running saves or restore in this environment.

A profile must contain a username, and one or more of the following options:

- **CLUSTER=***cluster-name* – A name of a cluster (group). This is used together with the username to identify the user account on the cluster. Specify "*" for all clusters.
- **NODE=***node-name* – A name of a node. This is used together with the username to identify the user account on the node. Specify "*" for all nodes.
- **PRIVILEGES=**(*privileges*[,...]) – A list of OpenVMS privileges. If specified, this overrides the default privileges obtained from the user's UAF record on the node or cluster.
- **RIGHTS=**(*right*[,...]) – A list of rights names. If specified, this overrides the default rights obtained from the user's UAF record on the node or cluster.

If the username is not specified, the existing username in the profile is assumed. The username must not contain a node or cluster name, and must be a valid OpenVMS user on the node or cluster specified.

If the profile is not specified, the following defaults are applied:

USER=ABS

CLUSTER=*

NODE=*

PRIVILEGES=NONE (will use ABS privileges on execution node)

RIGHTS=NONE (will use ABS rights on execution node)

If just a username is provided, the default node is the node in which the command is entered. The privileges and rights are obtained from the UAF file for the specified user on the specified node (or the command node if none).

User name can also be specified as a special keyword, <REQUESTER>. This means that the user profile in the associated save or restore request is used. This is the normal situation for user backups. The other profile fields are not used in this case.

/PROLOGUE="*string*"**/NOPROLOGUE (Default)**

The **/PROLOGUE** qualifier specifies a string to execute before an entire save or restore operations starts. The string may be up to 80 characters and must be enclosed in quotes.

By default, no prologue command is executed.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/RETRY_LIMIT=number
/NORETRY_LIMIT

The **/RETRY_LIMIT** qualifier specifies the number of retries performed if a save or restore operation fails. If you specify a retry limit of 0, no retries are attempted. **/NORETRY_LIMIT** means that there is no limit on the number of retries.

If not specified, the default value is 3.

Examples

The following command creates an environment `USER_SAVE_ENV`, specifying an action to record backup dates on files, and adding a notification to mail to user `SYSTEM` a normal level of detail at the completion of any operation. The profile name of `<REQUESTER>` specifies a user backup that will use the profile in the save or restore request:

```
$ MDMS CREATE ENVIRONMENT USER_SAVE_ENV -
_$ /ACTION=RECORD -
_$ /PROFILE=USER=<REQUESTER> -
_$ /NOTIFICATION=(MAIL=SYSTEM, TYPE=NORMAL, WHEN=COMPLETE)
```

The following command creates an environment `SYSTEM_SAVE_ENV`, specifying no operator assistance, a full listing, and a retry limit of 1000 at 2-minute intervals:

```
$ MDMS CREATE ENVIRONMENT SYSTEM_SAVE_ENV -
_$ /NOASSIST -
_$ /LIST=FULL -
_$ /INTERVAL=00:02:00
_$ /RETRY_LIMIT=1000
```

MDMS CREATE GROUP

MDMS CREATE GROUP — Creates a definition of a group of nodes in the MDMS domain.

Format

```
MDMS CREATE GROUP [group-name] [, ...]
```

Equivalent **STORAGE** Command: None.

Parameters

group-name

Specifies the name of the group. A group includes nodes that share a common storage device, or some other relationship. There is no limit to the number of groups that you may specify, and any node may appear in any number of groups. Groups may or may not be equivalent to clusters in your environment. Groups may be used instead of nodes in drive and jukebox definitions, and also as authorized or default users in pool definitions.

The maximum length of the group name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of group names may be entered.

Description

The **MDMS CREATE GROUP** command creates a new group definition in the MDMS database. When creating a new object record, the user can specify attribute values or allow for MDMS default value assignments. Alternatively, values can be inherited from a specified group using the **/INHERIT** qualifier.

Privileges

The request requires MDMS_CREATE_ALL.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/DESCRIPTION=*"text"*

Comments about the group. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/INHERIT=*group-name*

This qualifier allows you to specify an existing group record to inherit attribute values. All attributes may be inherited except for group name.

/NODES=(*node-name*[,...])
/NONODES

Specifies one or more distinct nodes that are members of the group. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace objects in the list, rather than adding them by default. The **/NONODES** qualifier removes all nodes.

When specifying nodes, the node names are the MDMS node names, which should have an equivalent node object (not the DECnet-Plus or TCP/IP fullnames).

/OWNER_NAME=*node::username*
/OWNER_NAME=*group::username*
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

Examples

The following command creates a group named **HOSER** and includes the nodes **TOOKUS**, **GREAT**, and **NORTH**:

```
$ MDMS CREATE GROUP HOSER /NODES=(TOOKUS, GREAT, NORTH)
```

The following command creates a new group that includes the nodes in the group `HOSER` and adds node `LABATT` to the node list. You might use this command if the group `HOSER` includes all nodes in an OpenVMS Cluster, and you want to manage a device shared between that cluster and node `LABATT` (which could be a member of a different OpenVMS Cluster):

```
$ MDMS CREATE GROUP MOUNTIE /INHERIT=HOSER -  
/NODES=LABATT /ADD
```

MDMS CREATE JUKEBOX

MDMS CREATE JUKEBOX — Creates a new jukebox definition in the MDMS database.

Format

```
MDMS CREATE JUKEBOX [jukebox-name] [, ...]
```

Equivalent **STORAGE** Command: None.

Parameters

jukebox-name

Specifies the name of the jukebox. The jukebox must be unique in the MDMS domain.

The maximum length of the jukebox name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of jukebox names may be entered.

Description

The **MDMS CREATE JUKEBOX** command creates a new jukebox definition in the MDMS database. MDMS supplies attribute values when creating a new object record. Alternatively, they can be inherited from a specified jukebox using the **/INHERIT** qualifier.

An MDMS jukebox may be controlled by the MRD subsystem if it accepts direct SCSI robotic operations. Each MRD-controlled jukebox contains a set of slots, drives, and robotics configured with a single robotic device name, regardless of whether the jukebox is a loader, a single tower library or a multi-tower library. A robot name must be associated with each MRD-controlled jukebox. For multi-tower configurations, a topology field may optionally be associated with the jukebox if the jukebox supports direct magazine moves.

Certain other types of jukeboxes, specifically silos manufactured by StorageTek, require the DCSC subsystem for control. With this model, each MDMS jukebox object maps to a Library Storage Module (LSM) containing a carousel of cells, drives, and robotics. One or more LSMs are contained in an Automated Cartridge System (ACS), and one or more ACS's are managed by a UNIX-based system called a library. For each DCSC-controlled jukebox, the library, ACS ID, and LSM ID must be specified. In addition, each LSM contains one or more Cartridge Access Points (CAPs) that are used to move volumes into and out of the jukebox (LSM). Each CAP may contain a different number of cells, so the CAP size should be specified for each CAP for optimal performance. A value of 40 is used if the CAP size is not specified for a particular CAP.

Privileges

The request requires MDMS_CREATE_ALL.

The **/STATE** qualifier also requires MDMS_SET_PROTECTED since this attribute is normally managed by MDMS. You should not modify this attribute unless you are trying to recover from an abnormal situation.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS=keyword

This qualifier defines the type of access to the jukebox, which can be one of the following keywords:

ALL (Default)	Supports local node/cluster/fibre access and remote (RDF) access
LOCAL	Supports local node/cluster/fibre access only
REMOTE	Supports remote (RDF) access only

/ACCESS_CONTROL=(USERNAME=user-id, ALLOW=(option[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ACS=*acs-id*

The **/ACS** qualifier specifies the Automated Cartridge System (ACS) identifier. The default value is zero. Each MDMS jukebox maps to one Library Storage Module (LSM), and requires the specification of the library, ACS and LSM identifiers. Valid for DCSC-controlled jukeboxes only.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/AUTOMATIC_REPLY (Default)**/NOAUTOMATIC_REPLY**

Specifies that MDMS automatically replies to all OPCOM messages that can be polled for completion on requests for this particular jukebox.

/CAP_SIZE=(*number*[,...])

For DCSC-controlled jukeboxes equipped with Cartridge Access Points (CAPs), this attribute specifies the number of cells for each CAP. The first number is the size for CAP 0, the next for CAP 1 etc. If a size is not specified for a CAP, a default value of 40 is used. Specifying the CAP size optimizes the movement of volumes to and from the jukebox by filling the CAP to capacity for each move operation. When specifying CAP sizes, the specified numbers always replace any previous sizes in the database. Valid for DCSC-controlled jukeboxes only.

/CONTROL=*keyword*

This qualifier specifies the robot control facility used to control the jukebox's robot. The valid keywords are:

MRD (Default)	The robot is controlled with the MRD facility
DCSC	The jukebox is a silo controlled by the DCSC facility (for StorageTek(R) Silos only)

/DESCRIPTION="*text*"

Defines comments about the object in the record. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the existing description, specify "".

/DISABLED

Places the jukebox in the disabled state. This prevents all drives in the jukebox from being selected and allocated for use. This takes effect immediately. However, any drives in use will continue to be used until they are deselected.

/ENABLED (Default)

Places the jukebox in the enabled state. This allows drives in the jukebox to be selected and allocated for use. This takes effect immediately. This is the default.

/GROUPS=(*group-name*[,...])
/NOGROUPS

Specifies the names of groups of nodes that share common access to this device. Normally, only one group is specified. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace objects in the list, rather than adding them by default. The **/NOGROUPS** qualifier removes all groups. If neither the **/NODES** nor **/GROUPS** qualifier is specified, then the node from which the request was issued is used as the node name. Groups can also be specified with the **/NODES** qualifier.

/INHERIT=*jukebox-name*

This qualifier allows you to specify an existing jukebox record to inherit attributes. MDMS supplies values for attributes you do not specify on creation. All attributes may be inherited except, for Jukebox name, Robot name, and State.

/LIBRARY=*library-id*

This qualifier specifies the library identifier for a silo. Valid values are 1,2,3,4, and the default is 1 when the jukebox is controlled by DCSC and 0 (not applicable) when controlled by MRD. Each MDMS jukebox maps to one Library Storage Module (LSM), and requires specification of the library, ACS and LSM identifiers.

/LOCATION=*location-name*
/NOLOCATION

This qualifier specifies the location of the jukebox, which is used when moving volumes into and out of the jukebox. If not specified, or if **/NOLOCATION** is specified, the default on-site location from the domain record is used as the jukebox location.

/LSM=*lsm-id*

The **/LSM** qualifier specifies the Library Storage Module (LSM) identifier. The default value is zero. Each MDMS jukebox maps to one Library Storage Module (LSM), and requires the specification of the library, ACS and LSM identifiers. Valid for DCSC-controlled jukeboxes only.

/NODES=(*node-name*[,...])
/NONODES

Specifies one or more nodes that can directly access the jukebox. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace objects in the list, rather than adding them by default.

The **/NONODES** qualifier removes all nodes.

If neither the **/NODES** nor **/GROUPS** qualifier is specified, then the node from which the request was issued is used as the node name.

/OWNER_NAME=*node::username*
/OWNER_NAME=*group::username*
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/ROBOT=robot-name

/NOROBOT

The **/ROBOT** qualifier defines the OpenVMS device name of the robot. Required for, and applicable to MRD-controlled jukeboxes only. Do not specify a node name in the robot name.

/SLOT_COUNT=number

The **/SLOT_COUNT** qualifier specifies the total number of slots in the entire jukebox. For any jukebox, either the slot count or topology must be specified. Valid for MRD-controlled jukeboxes only.

/STATE=keyword

This is a protected field that should be modified only to recover on error. Use the **LOAD**, **UNLOAD**, or **MOVE** commands to manipulate the state field under normal operation. The **/STATE** qualifier specifies the usage state of the jukebox. The keyword values are:

Available	The jukebox is available for use
In_use	The jukebox is in use

This qualifier requires the right MDMS_SET_PROTECTED.

/THRESHOLD=number

Specifies that an OPCOM message is output when the number of free volumes in the jukebox falls below the specified number. The default value is zero, which disables the feature. The OPCOM message is output on a node that can directly access the jukebox.

/TOPOLOGY=(TOWERS=(number[,...]), FACES=(number[,...]), LEVELS=(number[,...]), SLOTS=(number[,...]))

Specifies topology of jukebox, when a TL820-class jukebox is being used as a magazine. Valid for MRD-controlled jukeboxes only. The topology specification allows OPCOM messages to move

magazines to be specified with TOWER, FACE, LEVEL rather than slot range. The specification of topology is optional. For each tower in the configuration, a corresponding entry must also be placed in FACES LEVELS and SLOTS that reflects the configuration of that tower.

The tower numbers start at zero, and additional towers must be the next number in sequence (i.e. 0,1,2 etc). Other specifications are absolute counts of the entity being specified for each tower (i.e. the total number of faces, levels and slots in each tower).

For example, for a three-tower jukebox, each tower having 8 faces, the first tower having two levels and the other two towers having three levels, and support of 11-slot bin-packs, the topology specification would be:

```
/TOPOLOGY=(TOWERS=(0,1,2), /FACES=(8,8,8), LEVELS=(2,3,3),  
SLOTS=(11,11,11))
```

/USAGE=[NO]MAGAZINE

The **/USAGE=MAGAZINE** qualifier specifies that the jukebox is configured for magazines and that the movement of volumes may be performed using magazines. The **/USAGE=NOMAGAZINE** qualifier does not support magazine use. The default is NOMAGAZINE. You must specify **/USAGE=MAGAZINE** when defining the **/TOPOLOGY** attribute. Note that you can use the jukebox for non-magazine moves even when the usage is magazine, but the reverse is not true. Valid for MRD-controlled jukeboxes only.

Examples

The following command creates a jukebox JUKE_1 with 7 slots and controlled by robot \$1\$DUA512: and supports magazines. The control type is MRD by default, and the current node is used as the node name supporting the jukebox:

```
$ MDMS CREATE JUKEBOX JUKE_1 /SLOT_COUNT=7 -  
/ROBOT=$1$DUA512: /USAGE=MAGAZINE
```

The following command creates a jukebox JUKE_2 with 140 slots and controlled by robot \$1\$DUA600 in group COOKIE. The control type is MRD by default and magazines are not supported:

```
$ MDMS CREATE JUKEBOX JUKE_2 /SLOT_COUNT=140 -  
/ROBOT=$1$DUA600 /USAGE=NOMAGAZINE /GROUPS=COOKIE
```

The following command creates a DCSC-controlled jukebox JUKE_3 with a library number of 1, ACS ID of 0 and LSM ID of 0. In addition, there are two CAPS with sizes of 20 for CAP 0 and 30 for CAP 1. The control type is DCS meaning this is a silo, does not support magazines, and other default attribute values are supplied by MDMS:

```
$ MDMS CREATE JUKEBOX JUKE_3 /CONTROL=DCSC /LIBRARY=1 -  
/ACS=0 /LSM=0 /CAP_SIZE=(20,30)
```

The following command creates a multi-tower jukebox, controlled by MRD with robot name \$1\$DUA510:. Its topology is TOWER 0 with 8 faces, 3 levels, and 11 slots; TOWER 1 with 8 faces, 2 levels, and 11 slots:

```
$ MDMS CREATE JUKEBOX JUKE_4 /CONTROL=MRD -  
/ROBOT=$1$DUA510: /TOPOLOGY=(TOWERS=(0,1), FACES=(8,8), -  
LEVELS=(3,2), SLOTS=(11,11))
```


MDMS CREATE LOCATION

MDMS CREATE LOCATION — Creates a new location definition in the MDMS database.

Format

```
MDMS CREATE LOCATION [location] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

location

Specifies the name of the location.

The maximum length of the location is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of locations may be entered.

Description

The **MDMS CREATE LOCATION** command creates a new location definition in the MDMS database. MDMS supplies attribute values you do not define on creation. Alternatively, they can be inherited from a named location object using the **/INHERIT** qualifier.

Privileges

The request requires MDMS_CREATE_ALL.

Restrictions

None.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object

- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/DESCRIPTION="text"

Defines comments about the object in the record. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the existing description, specify "".

/INHERIT=location-name

This qualifier allows you to specify an existing location record to inherit default attributes. The default is that MDMS supplies the default values. All attributes may be inherited except for location name.

/LOCATION=location

/NOLOCATION (Default)

The **/LOCATION** qualifier allows you to specify a parent location, thus creating a location hierarchy. If there is no parent location, specify **/NOLOCATION**. Use parent locations to allow selection of volumes or drives in compatible locations. One location is compatible with another if it has a common parent location in the hierarchy. If you do not wish to utilize the compatible location feature, do not specify parent locations. Locations with common parents are most useful where the parents and siblings are in close proximity to one another (e.g. rooms 101 and 102, with parent location floor 1), and selection of volumes or drives from any of the locations is desired. Do not use parent locations across larger distances.

/OWNER_NAME=node::username

/OWNER_NAME=group::username

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/SPACES=(*range*)
/NOSPACES

The **/SPACES** qualifier defines individual spaces for volumes or magazines at the location. Spaces are alphanumeric strings of up to 8 characters. The spaces can be specified as a range—only a single range is supported.

When specifying a range, the first and last spaces in the range must have the same number of characters (as in the example), and there is a limit of 1000 spaces per location. The **/NOSPACES** qualifier removes all spaces.

Examples

The following command creates a shelf location for volumes with 100 spaces, named 40:S001 to 40:S100:

```
$ MDMS CREATE LOCATION SHELF_40 /SPACES=(40:S001-40:S100)
```

The following command creates a location named VAULT_1 with no spaces:

```
$ MDMS CREATE LOCATION VAULT_1
```

MDMS CREATE MAGAZINE

MDMS CREATE MAGAZINE — Creates a new magazine definition in the MDMS database.

Format

```
MDMS CREATE MAGAZINE [magazine-name] [, ...]
```

Equivalent STORAGE Command: **STORAGE ADD MAGAZINE**

Parameters

magazine-name

Specifies the name of the magazine.

The maximum length of the magazine name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of magazine names may be entered.

Description

The **MDMS CREATE MAGAZINE** command creates a new magazine definition in the MDMS database. Magazines are used for two reasons:

- For compatibility with previous versions of MDMS, where loader-type jukeboxes could only be supported using magazines.
- To support a set of volumes in a physical magazine whose placement is the same for all volumes in the magazine. In other words, if a volume is in a magazine, it is moved with all the other volumes in the magazine.

It is important to note that you no longer have to use magazines for loader-type jukeboxes. You may instead choose to treat volumes separately and move them into and out of jukeboxes individually.

Privileges

The request requires MDMS_CREATE_ALL.

Several protected qualifiers also require MDMS_SET_PROTECTED since these attributes are normally managed by MDMS. You should not modify these attributes unless you are trying to recover from an abnormal situation.

The **/JUKEBOX**, **/PLACEMENT**, **/POSITION**, and **/START_SLOT** are qualifiers that also require MDMS_SET_PROTECTED.

Restrictions

None.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/DESCRIPTION="*text*"

Defines comments about the object in the record. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters.

To clear the existing description, specify "".

/INHERIT=*magazine-name*

This qualifier allows you to specify an existing magazine record to inherit default attributes. The default is that MDMS supplies values you do not specify. All attributes may be inherited except for Magazine name, Jukebox name, Placement, Position, and Start slot.

/JUKEBOX=*jukebox-name*

This is a protected field that should only be modified to recover on error. Use the **MOVE MAGAZINE** command to set up the jukebox name under normal operations. This qualifier specifies the name of the jukebox in which the magazine resides. The maximum length of the jukebox name is 31 characters.

This qualifier requires the right MDMS_SET_PROTECTED.

/OFFSITE=(**[LOCATION=***location***],[NO]DATE[=***date***])**
/NOOFFSITE

This qualifier specifies the date that the magazine is to be taken off site and the off-site location. The location field is required when using the **/OFFSITE** qualifier if no location has been previously specified. The LOCATION keyword cannot be negated and if specified must point to a valid location object. Specify a VMS absolute or delta time in the date field. The NODATE keyword may be used to remove the date. This has the effect of disabling the schedule for the magazine, while retaining the location. To clear the off-site date specify **/NOOFFSITE**.

The default off-site location is that stored in the domain object.

/ONSITE=(**[LOCATION=***location***],[NO]DATE[=***date***])**
/NOONSITE

This qualifier specifies the date that the magazine is to be brought back on site and the on-site location. The location field is required when using the **/ONSITE** qualifier if no location has been previously specified. The LOCATION keyword cannot be negated and if specified must point to a valid location object. Specify a VMS absolute or delta time in the date field. The NODATE keyword may be used to remove the date. This has the effect of disabling the schedule for the magazine, while retaining the location. To clear the on-site date, specify **/NOONSITE**.

The default on-site location is that stored in the domain object.

/OWNER_NAME=*node::username*
/OWNER_NAME=*group::username*
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/PLACEMENT=keyword

This is a protected field that should only be modified to recover on error. Use the **MOVE MAGAZINE** command to set up the placement under normal operation. This qualifier defines the current placement of the magazine. Select one of the following options:

JUKEBOX
OFFSITE
ONSITE
MOVING

This qualifier requires the right MDMS_SET_PROTECTED.

/POSITION=position=(tower,face,level)

This is a protected field that should only be modified to recover on error. Use the **MOVE MAGAZINE** command to set up the position under normal operation. The **/POSITION** qualifier specifies the position in the jukebox that the magazine resides.

The tower, face and level specification represent the relative number of the tower, face and level, starting from 0. So for absolute jukebox slot of zero, the corresponding position is (0,0,0). The next position in the jukebox would be (0,0,1) and so on, according to the topology defined for the jukebox.

This qualifier requires the right MDMS_SET_PROTECTED.

/SLOT_COUNT=number

The **/SLOT_COUNT** qualifier specifies the number of slots in a magazine to store volumes. The default slot count is 32 which can be used for all magazines, but you should specify the actual value for best results.

/SPACES=(range)

/NOSPACES

This qualifier specifies the spaces in a location in which the magazine is stored when not in a jukebox. Spaces are alphanumeric strings of up to 8 characters. The **/NOSPACES** qualifier removes all spaces.

/START_SLOT=(number)

This is a protected field that should only be modified to recover on error. Use the **MOVE MAGAZINE** command to set up the start slot under normal operation. This qualifier specifies the starting jukebox slot when the magazine is placed in a jukebox.

This qualifier requires the right MDMS_SET_PROTECTED.

Examples

The following command creates a magazine called MYMAG01 with 7 slots, which is stored in location space 2 in location SHELF1 when not in a jukebox:

```
$ MDMS CREATE MAGAZINE MYMAG01 /SLOT_COUNT=7 -
```

```
/ONSITE=(LOCATION=SHELF1) /SPACE=2
```

The following command creates a magazine MAG002, with 32 slots, which is stored in on-site location HEADQUARTERS in spaces SA001-SA032, due to be moved off site on 01-Jan-2025 to location DPS, and back on site on 01-Jan-2026:

```
$ MDMS CREATE MAGAZINE MAG002 /SLOT_COUNT=32 -  
/SPACES=(SA001-SA032) /ONSITE=(LOCATION=HEADQUARTERS, -  
DATE=01-JAN-2026) /OFFSITE=(LOCATION=DPS, -  
DATE=01-JAN-2025)
```

The following command creates a magazine MAG003, with 32 slots, which is stored in space SPC001. When in a jukebox, the magazine resides in jukebox JUKE_1 in position tower 0, face 1, level 2. Note that jukebox and position should not normally be specified—rather, these are set up when moving the magazine into the jukebox with a **MOVE MAGAZINE** command:

```
$ MDMS CREATE MAGAZINE MAG003 /SLOT_COUNT=32 -  
/SPACES=SPC001 /JUKE=JUKE_1 /POSITION=(0,1,2)
```

MDMS CREATE MEDIA_TYPE

MDMS CREATE MEDIA_TYPE — Creates a new media type definition in the MDMS database.

Format

```
MDMS CREATE MEDIA_TYPE [media-type] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

media-type

Specifies the name of the media type.

The maximum length of the media type name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of media types may be entered.

Description

The **MDMS CREATE MEDIA_TYPE** command creates a new media type definition in the MDMS database. A media type definition consists of a required density, and optional compaction length and capacity, if applicable.

Privileges

The request requires MDMS_CREATE_ALL.

Restrictions

None.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/CAPACITY=*number*

The **/CAPACITY** qualifier specifies the capacity in megabytes of the tape. This is used by some MDMS clients to estimate end-of-tape conditions. By default, capacity is set to zero.

/COMPACTION (Default)
/NOCOMPACTION

The **/COMPACTION** qualifier specifies that the media type should use compaction when writing to tape. This is the default. If you do not wish to use compaction, then specify **/NOCOMPACTION**.

/DENSITY=*density*

Specifies a density string between 1 and 31 characters in length that the media type supports. Note that the COMP keyword for compaction should be specified in the **/COMPACTION** attribute, not density.

/DESCRIPTION="*text*"

Defines comments about the media type. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description is can range from 0 to 255 characters.

Specify "" to clear the description.

/INHERIT=media-type-name

This qualifier allows you to specify an existing media type record to inherit default attributes. The default is that MDMS supplies values you do not specify on creation. All attributes may be inherited except for media type name.

/LENGTH=length

The **/LENGTH** qualifier specifies the length of a 9-track magnetic tape, and is expressed in feet.

By default, length is set to zero.

/OWNER_NAME=node::username

/OWNER_NAME=group::username

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

Examples

The following command creates a new media type called TK85K_COMP with compaction enabled:

```
$ MDMS CREATE MEDIA_TYPE TK85K_COMP /COMPACTION
```

The following command creates a new media type called TAPE_800, a 9-track media type with 800 bpi density and a length of 2400 feet:

```
$ MDMS CREATE MEDIA_TYPE TAPE_800 /DENSITY=800 -  
/LENGTH=2400
```

The following command creates a new media type called TAPE_1600, a 9-track media type with 1600 bpi density and a length of 2400 feet (inherited from media type TAPE_800):

```
$ MDMS CREATE MEDIA_TYPE TAPE_1600 -  
/INHERIT=TAPE_800 /DENSITY=1600
```

MDMS CREATE NODE

MDMS CREATE NODE — Creates a new node definition in the MDMS node database.

Format

```
MDMS CREATE NODE [node-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

node-name

Specifies the name of the node. Do not append colons to the node name. The node name should be the DECnet Phase IV node name (i.e. logical SYSSNODE) if DECnet Phase IV is supported on the node.

Otherwise, it should be a unique name chosen by the MDMS administrator and it should match the SYSGEN parameter SCSNODE. If SCSNODE is not defined, the node name has to match the host or node name portion of the DECnet-Plus or TCP/IP fullname. If SCSNODE name is not defined and no network name is available, the server starts up with a name of "MDMS\$SERVER".

If DECnet-Plus and/or TCP/IP are used, the appropriate fullnames should be stored as attributes of the node. Do not use the node name to specify fullnames.

The maximum length of the node name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of node names may be entered.

Description

The **MDMS CREATE NODE** command creates a new node definition in the MDMS database. A node record must exist for each node that can act as a client to the database server. In addition, a node record must exist for each node capable of being a database server.

When creating a new object record, default attributes are supplied by MDMS. Alternatively, they can be inherited from a specified node using the **/INHERIT** qualifier.

Privileges

The request requires MDMS_CREATE_ALL.0

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/DATABASE_SERVER (Default) **/NODATABASE_SERVER**

The **/DATABASE_SERVER** qualifier means the node can be a database server, supporting failover operations. To be a database server, the node must have direct access to the MDMS Database files.

In addition, this node name should be added to the definition of the logical name MDMS\$DATABASE_SERVERS in SYS\$STARTUP:MDMS\$SYSTARTUP.COM on all nodes in the domain.

/DECNET_PLUS_FULLNAME=*node-fullname*

This qualifier allows you to specify the DECnet-Plus fullname for a node. The fullname may be up to 255 characters. If this node has a DECnet-Plus name defined by logical name SYS\$NODE_FULLNAME, then the DECNET_PLUS_FULLNAME has to be defined for this node and has to exactly match the DECnet-Plus name.

The DECNET_PLUS_FULLNAME has to be defined in order for this node to be fully enabled when the DECnet transport has been enabled and DECnet-Plus is running on the system. The fullname can be specified in upper or lower case.

/DESCRIPTION="*text*"

Defines comments about the node. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. Specify "" to clear the description.

/DISABLED

Places the node in the disabled state. This prevents the node from participating in the MDMS domain as either a server or a client. This takes effect immediately.

/ENABLED (Default)

Places the node in the enabled state. This allows the node to participate in MDMS operations. This takes effect immediately. This is the default.

/INHERIT=*node-name*

This qualifier allows you to specify an existing node record to inherit attributes. The default is that MDMS supplies attribute values you do not supply on creation. All attributes may be inherited except for Node name, DECnet_Plus fullname, and TCPIP fullname.

/LOCATION=*location***/NOLOCATION**

The **/LOCATION** qualifier specifies the location of the node, which is used when allocating volumes and drives. If not specified, or if **/NOLOCATION** is specified, the default on-site location from the domain record is used as the node location.

/OPCOM=(*class*[,...])**/NOOPCOM**

The **/OPCOM** qualifier adds the specified classes used for notifying operators. All OPCOM for devices on the node are sent to all specified classes on the node. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace classes in the list, rather than adding them by default. Specify **/NOOPCOM** to disable OPCOM notification. By default, the node acquires OPCOM classes from the domain record.

The following classes are valid:

CARDS	NETWORK	OPER6	OPER12
CENTRAL	OPER1	OPER7	PRINTER
CLUSTER	OPER2	OPER8	REPLY
DEVICES	OPER3	OPER9	SECURITY
DISKS	OPER4	OPER10	SOFTWARE
LICENSE	OPER5	OPER11	TAPES

/OWNER_NAME=*node::username***/OWNER_NAME=*group::username*****/NOOWNER_NAME**

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an

owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/TCPIP_FULLNAME=*node-fullname[:low-port-high-port]*

This qualifier allows you to specify the TCP/IP full name for a node. The full name may be up to 255 characters. If this node has a TCP/IP name defined by the logical name *"*INET_HOST"*, then the **TCPIP_FULLNAME** has to be defined and has to exactly match the full IP name as *"<INET_HOST>.<INET_DOMAIN>"*. For *INET_DOMAIN*, see the logical name *"*INET_DOMAIN"*. The **TCPIP_FULLNAME** has to be defined in order for this node to be fully enabled when the TCP/IP transport has been enabled. The fullname can be specified in upper or lower case.

The *low-port* and *high-port* numbers specify the range of TCP/IP port numbers used by the server to listen for incoming requests. The default is 2501-2510. If this conflicts with other applications, a new range above 1023 can be specified. The range should contain at least 10 port numbers for the MDMS server to select one at a time.

Note that the MDMS GUI requires TCP/IP running on all GUI nodes, and on the MDMS server nodes to which the GUI may connect.

/TRANSPORT=(*keyword*[,...])

Specifies the network transports to be used, as a prioritized ordered list. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace objects in the list, rather than adding them by default.

Enter one or more of:

- **DECNET** – Listen to incoming requests from other MDMS servers on DECnet Phase IV and DECnet-Plus
- **TCPIP** – Listen to incoming requests from other MDMS servers on TCP/IP

Setting a new transport will automatically start the listener for this transport on the database server node. Likewise, removing a transport will take place within 10 seconds on the database server node. For client nodes, transport changes will take place the next time network connections time out (usually within 10 minutes). If the change needs to take place immediately, the client node server process must be restarted.

The node name and/or the node full names have to be set accordingly for a transport to work correctly.

Examples

The following command creates a new node definition named `TABLES` with a location, transport protocols, and OPCOM classes. The node supports DECnet (node name `TABLES`) and TCP/IP with a fullname of `TABLES.CXO.DEC.COM`:

```
$ MDMS CREATE NODE TABLES /LOCATION=COMPUTER_LAB_1 -  
/TRANSPORT=(DECNET,TCP/IP) /OPCOM=(CENTRAL,TAPES) -  
/TCP/IP_FULLNAME=TABLES.CXO.DEC.COM
```

The following command creates a new node named `CHAIRS`, which inherits location, OPCOM classes, and transport definitions from node `TABLES`, with specified DECnet-Plus and TCP/IP full names. The TCP/IP connections may be received over ports 3000–3050:

```
$ MDMS CREATE NODE CHAIRS /INHERIT=TABLES -  
/DECNET_PLUS_FULLNAME=DEC:CHAIRS.CXO.DEC.COM -  
/TCP/IP_FULLNAME=CHAIRS.CXO.DEC.COM:3000-3050
```

MDMS CREATE POOL

MDMS CREATE POOL — Creates a new pool definition in the MDMS database.

Format

```
MDMS CREATE POOL [pool-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

pool-name

Specifies the name of the pool.

The maximum length of the pool name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of pool names may be entered.

Description

The **MDMS CREATE POOL** command creates a new pool definition in the MDMS pool database. A pool definition consists of a list of authorized users, and users for whom the pool is the default pool. If a user is listed in either list, he/she is authorized for the pool.

Privileges

The request requires `MDMS_CREATE_ALL`.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/AUTHORIZED_USERS=(*node/group-name::username*[,...])

/NOAUTHORIZED_USERS

Specifies one or more distinct users to the pool specified by node or group name and user name. Only authorized or default users can allocate volumes belonging to the pool. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace users in the list, rather than adding them by default.

/DEFAULT_USERS=(*node/group-name::username*[,...])

/NODEFAULT_USERS

Specifies one or more distinct users to the pool as the users' default pool. Only authorized or default users can allocate volumes belonging to the pool. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace users in the list, rather than adding them by default. A particular *node/group::user* combination should only be defined with the **/DEFAULT** qualifier for one pool.

/DESCRIPTION=*text*

Defines comments about the pool. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. Specify "" to clear the description.

/INHERIT=*pool-name*

This qualifier allows you to specify an existing pool record to inherit attributes. The default is that MDMS supplies values you do not supply on creation. All attributes except pool name are inherited.

/OWNER_NAME=*node::username***/OWNER_NAME=*group::username*****/NOOWNER_NAME**

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/THRESHOLD=*number*

Specifies that an OPCOM message is output when the number of free volumes in the pool falls below the specified number. The default value is zero, which disables the feature. The OPCOM message is output on the database server node.

Examples

The following command creates a pool called TEST_POOL with one authorized user:

```
$ MDMS CREATE POOL TEST_POOL/AUTHORIZE=COOKIE::ABS
```

The following command adds default users to pool JIMS_POOL. The names before the double colons are a mixture of node names and group names:

```
$ MDMS CREATE POOL JIMS_POOL /DEFAULT=(OREO::JIM, -
```



```
CRUMBS::JIM, DSORDS::JIM) /ADD
```

MDMS CREATE RESTORE

MDMS CREATE RESTORE — Creates a new restore definition in the MDMS database. Restores were formerly known as restore requests or restore transactions. Restore is the new common name.

Format

```
MDMS CREATE RESTORE [restore-name] [, ...]
```

Equivalent ABS Command: **RESTORE**

Parameters

restore-name

Specifies the name of the restore.

The maximum length of the restore name is 39 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of restore names may be entered.

Description

The **MDMS CREATE RESTORE** command creates a new restore definition in the MDMS database. Restores are used to copy previously-saved data from an archive back to online disk storage.

When creating a new object record, MDMS supplies default values on attributes you do not specify. Alternatively, they can be inherited from a specified restore using the **/INHERIT** qualifier. In addition, you can create a restore by inheriting from a save request.

The data specifications for restores can be supplied in one of two ways:

- Using the **/INCLUDE**, **/EXCLUDE**, **/DATA_SELECT_TYPE**, and **/SOURCE_NODE** qualifiers. These qualifiers can be used to specify a set of data specifications of a single data type that can be restored. These are not positional qualifiers. This option is recommended for most restores, and this information is stored in a default selection that is mapped to the restore by MDMS. The default selection name is *restore-name_REST_SEL_DEF*.

Note

In cases where the include specification in the Save request had a directory tree structure, maintain the same during the Restore operation by specifying the wildcard "*" in the include specification of the Restore request. For example:

```
/Include = TPRD:[TPRD.APPFILES*]*.*;*
```

If you do not use the wildcard "*" or specify a wildcard other than "*" in the include specification, then all the files are automatically restored to the root directory. For example:

```
/Include = TPRD:[TPRD.APPFILES]*.*;*  
/Include = TPRD:[TPRD.APPFILES%]*.*;*
```

- For more complex data specifications, you can create your own selections and map them to the restore using the **/SELECTIONS** qualifier. This options gives you more granularity in data selection, and allows you to have mixed data select types in a single restore. Most restores are **ONE_TIME_ONLY** (the default) and do not need to be scheduled. However, restores can also be regularly scheduled in the same way as saves. If you wish to use one of the supplied frequency options, MDMS creates a schedule and initializes it to appropriate values. The schedule for any restore is called *restore-name_REST_SCHED*.

Alternatively, if you wish to define a schedule not handled in the supplied frequencies, you can use one of the following options:

- For scheduler types **INTERNAL** and **EXTERNAL**, you can define **/FREQUENCY=CUSTOM** and modify the schedule to your own specification.
- For scheduler type **SCHEDULER**, you can define **/FREQUENCY=EXPLICIT** and define a string to pass the explicit schedule to the scheduler in the qualifier **/EXPLICIT_INTERVAL**.

Note

When restoring a full disk volume to the destination that has disk quotas enabled, some manual intervention will be required to rebuild the *quota.sys* file after the restore completes.

The following command needs to be issued to correctly rebuild the "quota.sys" file:

```
{MCR SYSMAN DISK REBUILD/DEVICE=[device-name]}
```

where *device-name* is the name of the disk where files will be restored.

Privileges

The request requires **MDMS_CREATE_ALL**.

Restrictions

This command is restricted to nodes with an ABS license.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The **/DELETE_INTERVAL** and **/KEEP** qualifiers are mutually exclusive.

The **/BEFORE_DATE**, **/DATE_ARCHIVED**, and **/SINCE_DATE** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/ARCHIVE=(*archive-name*[,*archive-name*])

The **/ARCHIVE** qualifier allows you to specify one or two archives associated with the restore. The first archive name is applicable to full and selective threads in restores. If specified, the second archive name is associated with incremental threads in restores. If a second name is not specified, incremental threads use the first archive name. For any thread, the catalog name from the first archive is always used. At least one archive name must be supplied for a restore. If an archive is not specified, the archive "SYSTEM_BACKUPS" is used by default.

/BASE_DATE=*date*

/NOBASE_DATE

The base date is used as a starting point for restores that have a repeating schedule frequency. Subsequent schedules are applied by adding the multiples of the frequency to the base date. If the restore is not to be repeated, then no base date is needed.

By default, if a base date is not specified but a start date is, the base date is set to the start date. If neither is specified, the base date is not set and the restore is not scheduled for execution.

/BEFORE_DATE=*date*

/NOBEFORE_DATE

You can select files for restore based on the date archived before a certain date. If specified, the restore will apply to the latest date archived before the specified date. **/NOBEFORE_DATE** clears the before date and uses other date criteria to select files, as follows:

- **DATE_ARCHIVED** – The latest file saved on the specified date
- **SINCE_DATE** – The first file saved after the specified date
- No date – The latest copy of the file

/CATALOG=(NAME=*catalog-name*, [NO]NODES=(*node*[,...]))

/NOCATALOG (Default)

The **/CATALOG** qualifier specifies the name of the catalog used by the restore, and a list of nodes and groups that can directly access the catalog. This catalog overrides the catalog defined in the first archive specified in the restore. If you specify **/NOCATALOG**, then the catalog in the first restore archive is used.

Specify a local catalog name representing a catalog object in the MDMS database. With the **NODES** keyword, you may specify nodes, groups or a combination thereof. If you specify **NONODES**, the local execution node on which the restore is performed is used as the catalog node.

The default is **/NOCATALOG**.

/DATA_SELECT_TYPE=keyword

The **/DATA_SELECT_TYPE** qualifier specifies the type of data to restore in the default selection. Select one of the following options:

- **VMS_FILES**
- **VMS_SAVESET**

If not specified, MDMS attempts to determine the data select type based on the include file syntax. If MDMS cannot determine this, **VMS_FILES** is used. The data select type specification is stored in the default selection.

/DATE_ARCHIVED=date

/NODATE_ARCHIVED

You can select files for restore based on the date archived. If specified, the restore will apply to the latest copy of the data on the date archived. **/NODATE_ARCHIVED** clears the date archived and uses other date criteria to select files, as follows:

- **BEFORE_DATE** – The latest file saved before the specified date
- **SINCE_DATE** – The first file saved after the specified date
- No date – The latest copy of the file

/DELETE_INTERVAL=delta-time

The **/DELETE_INTERVAL** qualifier specifies the duration of certain restore requests after their start of execution. The qualifier is applicable only to restores with a scheduling frequency of **ONE_TIME_ONLY**. For these restores, the restore object (and any associated schedules and selections) are deleted after the delete interval has passed following the start of the restore execution. If you do not wish the restore to be deleted, specify **/KEEP**.

If not specified, the delete interval is 03-00:00:00 (3 days) for restores of frequency **ONE_TIME_ONLY**, and 0 for all other frequencies.

/DESCRIPTION=*text*

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/DESTINATION=*disk:dir*

The **/DESTINATION** qualifier specifies the target destination disk/directory location for the restore. Specify a disk and directory combination. If destination is not specified, then restore data is restored to the original disk location from where the data was saved. To clear the destination, enter "".

The default is **/NODESTINATION**, and the original save destination is used.

Restoring a Bound Volume Set:

You must enter the disk or file name exactly as you entered it on the save request, and you must enter the **/DESTINATION** qualifier that specifies the list of the disk devices in the bound volume set (a comma-separated list of disk names enclosed with in quotation marks). The disk name specified in the **/DESTINATION** qualifier can be a logical name or a physical device name. The logical name must be defined before creating/executing the restore request. For example:

```
ABS> RESTORE /NAME=BOUND_RESTORE DISK$USER1:/OBJECT_TYPE=VMS_FILES_ABS  
/DESTINATION="DISK$USER1:,DISK$USER2:,DISK$USER3:"
```

The equivalent MDMS command is as follows:

```
MDMS> CREATE RESTORE BOUND_RESTORE_MDMS  
/DESTINATION="DISK$USER1:,DISK$USER2:,DISK$USER3:"
```

Before executing the restore request, you must either dismount these output disks or mount them using the **/FOREIGN** qualifier.

However, if you perform a restore operation on a bound volume set, you must specify each disk in the bound volume set in either the Restore option (using the ABS GUI) or using the **/DESTINATION** qualifier (using the DCL command **ABS RESTORE**).

Concealed Logical Names:

If the data to be restored has a concealed logical name as part of the data include specification, the concealed logical name must be a system-wide logical name or physical device name. Restore request translates the logical name to the first concealed logical name found. However, if this concealed logical name is in the process table, access to the logical name is not available to ABS, and the restore operation fails.

/ENVIRONMENT=*environment-name*

The **/ENVIRONMENT** qualifier specifies the environment object to use for the restore. Specify an environment name in the MDMS database.

If no environment is specified, one of the following defaults is used, in order of priority:

1. The first archive name suffixed by "_ENV"

2. "DEFAULT_ENV"

/EPILOGUE="string"
/NOEPILOGUE (Default)

The **/EPILOGUE** qualifier specifies a string to execute after each restore thread (i.e. each include list item specification in the restore). The string may be up to 80 characters and must be enclosed in quotes.

By default, no epilogue command is executed.

/EXCLUDE=(data-specification[,...])
/NOEXCLUDE (Default)

The **/EXCLUDE** qualifier is used to exclude files that otherwise would be included with the **/INCLUDE** qualifier. This is used to exclude files for save or restore when a more general wildcard specification is specified for the include. The exclude specification is stored in the default selection.

By default, no files are excluded.

/EXECUTION_NODES=(node-name[,...])

The **/EXECUTION_NODES** qualifier specifies one or more node names on which the restore request can be executed. Any one of the nodes can be the execution node for the request. **/GROUPS** may also be specified.

If no nodes or groups are specified, the include specification uses the node on which the restore was created.

/EXPLICIT_INTERVAL="string"
/NOEXPLICIT_INTERVAL (Default)

When using scheduler type **SCHEDULER** you can specify the scheduling frequency using **/EXPLICIT_INTERVAL**. Enter a string in a format known to the scheduler.

/NOEXPLICIT_INTERVAL clears the explicit interval.

This qualifier is only applicable when the **/FREQUENCY=EXPLICIT** qualifier is specified.

By default, no explicit interval is defined.

/FREQUENCY=keyword

The **/FREQUENCY** qualifier specifies the scheduling frequency for this restore. Select one of the following options:

- **ONE_TIME_ONLY (Default)** – Scheduled one time only based on the base date or start date. This restore cannot be rescheduled, and the restore will automatically be deleted when the delete interval expires. This is the default.
- **ON_DEMAND** – Scheduled on demand when a start date (or base date) is entered. Can be rescheduled by entering a new start date (or base date).
- **DAILY** – Scheduled on a daily basis at the time contained within the base date. For example: 21-JAN-2001, 22-JAN-2001...

- WEEKLY – Scheduled on a weekly basis on anniversaries of the base date day (MON-FRI) and time. For example: 21-JAN-2001, 28-JAN-2001...
- BIWEEKLY – Scheduled on a biweekly (once every two weeks) basis on anniversaries of the base date day (MON-FRI) and time. For example: 21-JAN-2001, 04-FEB-2001...
- SEMI_MONTHLY – Scheduled twice a month based on the base date (1-31) and time, and either two weeks before or after the date. For example: 21-JAN-2001, 07-FEB-2001... +CHECK++
- MONTHLY – Scheduled monthly based on the base date (1-31) and time and subsequent monthly anniversaries. For example: 21-JAN-2001, 21-FEB-2001...
- QUARTERLY- Scheduled once every three months on the base date (1-31) and time, and three-month anniversaries thereafter. For example: 21-JAN-2001, 21-APR-2001...
- SEMI_ANNUALLY – Scheduled once every six months on the base date (1-31) and time, and six-month anniversaries thereafter. For example: 21-JAN-2001, 21-JUL-2001...
- ANNUALLY – Scheduled once every year on the base date (1-31) and time, and twelvemonth anniversaries thereafter. For example: 21-JAN-2001, 21-JAN-2002...
- NEVER – Cannot be scheduled. This option can be used as a placeholder. The frequency must be changed to something else before the restore can execute.
- EXPLICIT – When using scheduler type SCHEDULER, this option utilizes the string in EXPLICIT_INTERVAL to determine the scheduling frequency.
- CUSTOM – When using scheduler types INTERNAL or EXTERNAL, this option utilizes a user-defined schedule object, which has the frequency defined internally. Use this option if none of the above scheduling frequencies is appropriate.

/GROUPS=(*group-name*[,...])

/NOGROUPS

Specifies the names of groups of execution nodes on which this restore can be run. Usually, only one group is specified. If neither the **/EXECUTION_NODES** nor **/GROUPS** qualifier is specified, then the node from which the command was issued is used as the node name.

/INCLUDE=(*data-specification*[,...])

/NOINCLUDE (Default)

The **/INCLUDE** qualifier allows you to enter a list of up to 24 data specifications to restore. The data specifications can be OpenVMS disks or files. However, all specifications must be of the same data select type, as specified in the **/DATA_SELECT_TYPE** qualifier.

The include, exclude, and source node specifications, together with the data select type, are stored in a default selection object that has the name *restore-name*_REST_SEL_DEF.

This selection object can be modified independently to add such refinements as exclude lists, and additional selections can be added to any restore.

/NOINCLUDE removes the include file specification.

By default, there is no include specification.

/INCREMENTAL**/NOINCREMENTAL (Default)**

The **/INCREMENTAL** qualifier specifies that the restore should be an incremental restore. If not specified, the restore will be a full restore or selective restore based on the data-specification in the associated include specification or selections. **/NOINCREMENTAL** clears the incremental specification.

The default is **/NOINCREMENTAL**.

/INHERIT=restore-name**/INHERIT=save-name**

This qualifier allows you to specify an existing restore record from which the new object record inherits attribute values. MDMS supplies default values if you specify none. All attributes may be inherited with the exception of Restore Name.

For restores only, you can also inherit most attributes from a corresponding save. This is most useful if you want to restore everything that was saved in a specific save request. All attributes can be inherited except for the following:

- Restore Name
- Before Date
- Catalog
- Date Archived
- Destination
- Frequency
- Owner Name
- Since Date
- Start Time

With both types of inherit, the show only attributes of Job Number and Transaction Status are not inherited.

/KEEP

The **/KEEP** qualifier indicates that the restore is not to be automatically deleted, and clears the delete interval.

/OWNER_NAME=node::username**/OWNER_NAME=group::username****/NOOWNER_NAME**

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/PROLOGUE="string"
/NOPROLOGUE (Default)

The **/PROLOGUE** qualifier specifies a string to execute before each restore thread (i.e. each include list item specification in the restore). The string may be up to 80 characters and must be enclosed in quotes.

By default, no prologue command is executed.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/SELECTIONS=(selection,...)
/NOSELECTIONS

When using the default selection, this qualifier is not needed, as MDMS adds it automatically.

However, if you want to add your own selections to this restore, you add them with the **/SELECTION** qualifier. By default a selection is added, but you can also remove selections with the **/REMOVE** qualifier, and replace selections with the **/REPLACE** qualifier. **/NOSELECTIONS** removes all selections from the restore.

/SEQUENCE_OPTION=keyword

The **/SEQUENCE_OPTION** qualifier defines the type of sequencing used during the restore operation. Select one of the following options:

- **SEQUENTIAL** – Performs each restore thread sequentially. Each restore thread must fully complete before the next one starts.
- **OVERLAPPED** – Starts the next restore thread as soon as the backup drive or device is available for use.

The default is **SEQUENTIAL**.

/SINCE_DATE=date
/NOSINCE_DATE

You can select files for restore based on the date archived after a certain date. If specified, the restore will apply to the first date archived after the specified date. You can specify date as an

absolute date, as a combination of absolute and delta times, or as one of the following keywords: TODAY (default), TOMORROW, or YESTERDAY.

/NOSINCE_DATE clears the since date and uses other date criteria to select files, as follows:

- **BEFORE_DATE** – The latest file saved before the specified date
- **DATE_ARCHIVED** – The latest file saved on the specified date
- **No date** – The latest copy of the file

/SKIP_TIME=delta-time

/NOSKIP_TIME (Default)

It is possible to run a restore at any time, including times when it is not normally scheduled. You may, for example, want to run a restore three hours early, or you may want to run an extra restore for some reason.

By default, if you run a restore at an unscheduled time, MDMS will also run it at its regularly scheduled time as well. If you do not want this, then enter a skip time when starting the restore at the non-scheduled time. The skip time will be added to the start time, and MDMS will not schedule the restore until at least after the skip_time value.

The default is **/NOSKIP_TIME**.

/START_DATE[=date]

/NOSTART_DATE

The **/START_DATE** qualifier is used to schedule a restore to start at an unscheduled time. The **/BASE_DATE** is used to specify the normal scheduled times. By default, the start date is set to the base date if not separately specified.

The date value for the **/START_DATE** is optional. If not specified, the default is now.

/NOSTART_DATE clears the next start date, and the restore will not be scheduled.

Examples

The following commands creates a restore REST1, which uses archive REST1_ARC and environment REST1_ENV. The data to restore is NOD001::USER1\$: [SMITH...] *.*; * and NOD001::USER2\$: [JONES...] *.*; *, which will be restored to their original locations. The data select type is VMS_FILES. The **/START** qualifier starts the restore immediately, and the restore is a ONE_TIME_ONLY:

```
$ MDMS CREATE RESTORE REST1 -
_$ /ARCHIVE=REST1_ARC -
_$ /ENVIRONMENT=REST1_ENV -
_$ /INCLUDE=(USER1$: [SMITH...] *.*; *, -
_$ USER2$: [JONES...] *.*; *)
_$ /EXECUTION_NODES=NOD001 -
_$ /START
```

The following command creates a restore REST2, whose data selections are contained in user-created selection objects SELECT1, SELECT2, and SELECT3. This restore is regularly scheduled on a

monthly basis on the first of each month at 1:00 am. The default archive SYSTEM_BACKUPS and environment SYSTEM_BACKUPS_ENV are used:

```
$ MDMS CREATE RESTORE REST2 -  
_$/SELECTIONS=(SELECT1, SELECT2, SELECT3) -  
_$/FREQUENCY=MONTHLY -  
_$/BASE_DATE=01-JAN-2001:01:00:00
```

MDMS CREATE SAVE

MDMS CREATE SAVE — Creates a new save definition in the MDMS database. Saves were formerly known as save requests, archive requests or archive transactions. Save is the new common name.

Format

```
MDMS CREATE SAVE [save-name] [, ...]
```

Equivalent ABS Command: **SAVE**

Parameters

save-name

Specifies the name of the save.

The maximum length of the save name is 39 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of save names may be entered.

Description

The **MDMS CREATE SAVE** command creates a new save definition in the MDMS database. Saves are used to copy online disk data to an archive for the purposes of data safety (backup), disaster recovery, or long term off-line archival.

When creating a new object record, MDMS supplies default values on attributes you do not specify. Alternatively, they can be inherited from a specified SAVE using the **/INHERIT** qualifier.

The data specifications for saves can be supplied in one of two ways:

- Using the **/INCLUDE**, **/EXCLUDE**, **/DATA_SELECT_TYPE**, and **/SOURCE_NODE** qualifiers. These qualifiers can be used to specify a set of data specifications of a single data type that can be saved. These are not positional qualifiers. This option is recommended for most saves, and this information is stored in a default selection that is mapped to the save by MDMS. The default selection name is *save-name_SAVE_SEL_DEF*.
- For more complex data specifications, you can create your own selections and map them to the save using the **/SELECTIONS** qualifier. This options gives you more granularity in data selection, and allows you to have mixed data select types in a single save. Saves are normally scheduled to execute at regular intervals according to your site-specific needs. If you wish to use one of the supplied frequency options, MDMS creates a schedule and initializes it to appropriate values. The schedule for any save is called *save-name_SAVE_SCHED*.

Alternatively, if you wish to define a schedule not handled in the supplied frequencies, you can use one of the following options:

- For scheduler types **INTERNAL** and **EXTERNAL**, you can define **/FREQUENCY=CUSTOM** and modify the schedule to your own specification.
- For scheduler type **SCHEDULER**, you can define **/FREQUENCY=EXPLICIT** and define a string to pass the explicit schedule to the scheduler in the qualifier **/EXPLICIT_INTERVAL**.

Privileges

The request requires **MDMS_CREATE_ALL**.

Restrictions

This command is restricted to nodes with an ABS license. The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The **/DELETE_INTERVAL** and **/KEEP** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

If you are creating a save request for a bound volume set, enter only the volume set name or the first disk device name in the bound volume set. ABS will recognize each disk name assigned to the bound volume set.

Note

If you are restoring a bound volume set, you must enter each disk name in the bound volume set. See [MDMS CREATE RESTORE](#) for details about restoring a bound volume set.

If the include specification contains a concealed logical name, the concealed logical name must be a system-wide logical name or physical device name.

ABS will not translate concealed logical names defined process wide. If Save or Restore requests contain the concealed logical names, then logical names will be translated to the first concealed logical name found in the system table. However, if this concealed logical name is in the process table, access to the logical name is not available to ABS, and the save or restore operation will fail.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- **SET** – Allows setting the attributes of the object
- **SHOW** – Allows showing the object
- **DELETE** – Allows deleting the object

- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/ARCHIVE=(*archive-name*[,*archive-name*])

The **/ARCHIVE** qualifier allows you to specify one or two archives associated with the save. The first archive name is applicable to full and selective threads in saves. If specified, the second archive name is associated with incremental threads in saves. If a second name is not specified, incremental threads use the first archive name. For any thread, the catalog name from the first archive is always used. At least one archive name must be supplied for a save.

If an archive is not specified, the archive "SYSTEM_BACKUPS" is used by default.

/BASE_DATE=*date* **/NOBASE_DATE**

The base date is used as a starting point for saves that have a repeating schedule frequency. Subsequent schedules are applied by adding the multiples of the frequency to the base date. If the save is not to be repeated, then no base date is needed.

By default, if a base date is not specified but a start date is, the base date is set to the start date. If neither is specified, the base date is not set and the save is not scheduled for execution.

/DATA_SELECT_TYPE=*keyword*

The **/DATA_SELECT_TYPE** qualifier specifies the type of data to save in the default selection. Select one of the following options:

- VMS_FILES
- VMS_SAVESET

If not specified, MDMS attempts to determine the data select type based on the include file syntax. If MDMS cannot determine this, VMS_FILES is used. The data select type specification is stored in the default selection.

/DELETE_INTERVAL=*delta-time*

The **/DELETE_INTERVAL** qualifier specifies the duration of certain saves after their start of execution. The qualifier is applicable only to saves with a scheduling frequency of **ONE_TIME_ONLY**.

For these saves, the save object (and any associated schedules and selections) are deleted after the delete interval has passed following the start of the save execution. If you do not wish the save to be deleted, specify **/KEEP**.

If not specified, the delete interval is 03-00:00:00 (3 days) for saves of frequency **ONE_TIME_ONLY**, and 0 for all other frequencies.

/DESCRIPTION="*text*"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/ENVIRONMENT=*environment-name*

The **/ENVIRONMENT** qualifier specifies the environment object to use for the save. Specify an environment name in the MDMS database.

If no environment is specified, one of the following defaults is used, in order of priority:

1. The first archive name suffixed by "_ENV"
2. "DEFAULT_ENV"

/EPILOGUE="*string*"**/NOEPILOGUE (Default)**

The **/EPILOGUE** qualifier specifies a string to execute after each save thread (i.e. each include list item specification in the save). The string may be up to 80 characters and must be enclosed in quotes.

By default, no epilogue command is executed.

/EXCLUDE=(*data-specification*[,...])**/NOEXCLUDE (Default)**

The **/EXCLUDE** qualifier is used to exclude files that otherwise would be included with the **/INCLUDE** qualifier. This is used to exclude files for save or restore when a more general wildcard specification is specified for the include. The exclude specification is stored in the default selection.

By default, no files are excluded.

/EXECUTION_NODES=(*node-name*[,...])

The **/EXECUTION_NODES** qualifier specifies one or more node names on which the save request can be executed. Any one of the nodes can be the execution node for the request. **/GROUPS** may also be specified.

If no nodes or groups are specified, the include specification uses the node on which the save was created.

/EXPLICIT_INTERVAL="string"

/NOEXPLICIT_INTERVAL (Default)

When using scheduler type SCHEDULER, you can specify the scheduling frequency using **/EXPLICIT_INTERVAL**. Enter a string in a format known to the external scheduler. **/NOEXPLICIT_INTERVAL** clears the explicit interval.

This qualifier is only applicable when the **/FREQUENCY=EXPLICIT** qualifier is specified.

By default, no explicit interval is defined.

/FREQUENCY=keyword

The **/FREQUENCY** qualifier specifies the scheduling frequency for this save. Select one of the following options:

- **ONE_TIME_ONLY (Default)** – Scheduled one time only based on the base date or start date. This save cannot be rescheduled, and the save will automatically be deleted when the delete interval expires. This is the default.
- **ON_DEMAND** – Scheduled on demand when a start date (or base date) is entered. Can be rescheduled by entering a new start date (or base date).
- **DAILY** – Scheduled on a daily basis at the time contained within the base date. For example: 21-JAN-2001, 22-JAN-2001...
- **DAILY_FULL_WEEKLY** – Scheduled on a daily basis, with a full save on the base date and weekly anniversaries, and incremental saves on other days. For example: 21-JAN-2001, 22-JAN-2001...
- **WEEKLY** – Scheduled on a weekly basis on anniversaries of the base date day (MON-FRI) and time. For example: 21-JAN-2001, 28-JAN-2001...
- **BIWEEKLY** – Scheduled on a biweekly (once every two weeks) basis on anniversaries of the base date day (MON-FRI) and time. For example: 21-JAN-2001, 04-FEB-2001...
- **SEMI_MONTHLY** – Scheduled twice a month based on the base date (1-31) and time, and either two weeks before or after the date. For example: 21-JAN-2001, 07-FEB-2001... +
+CHECK++
- **MONTHLY** – Scheduled monthly based on the base date (1-31) and time and subsequent monthly anniversaries. For example: 21-JAN-2001, 21-FEB-2001...
- **QUARTERLY** – Scheduled once every three months on the base date (1-31) and time, and three-month anniversaries thereafter. For example: 21-JAN-2001, 21-APR-2001...
- **SEMI_ANNUALLY** – Scheduled once every six months on the base date (1-31) and time, and six-month anniversaries thereafter. For example: 21-JAN-2001, 21-JUL-2001...
- **ANNUALLY** – Scheduled once every year on the base date (1-31) and time, and twelvemonth anniversaries thereafter. For example: 21-JAN-2001, 21-JAN-2002...
- **LOG_2** – Scheduled daily based on a full and incremental algorithm to optimize save/restore times. For example: 21-JAN-2001, 22-JAN-2001... (see the [VSI Archive Backup System for OpenVMS User Guide \[https://docs.vmssoftware.com/vsi-abs-user-guide/\]](https://docs.vmssoftware.com/vsi-abs-user-guide/))

- LOG_3 – Scheduled daily based on a full and incremental algorithm to optimize save/restore times. For example: 21-JAN-2001, 22-JAN-2001... (see the [VSI Archive Backup System for OpenVMS User Guide](https://docs.vmssoftware.com/vsi-abs-user-guide/) [https://docs.vmssoftware.com/vsi-abs-user-guide/])
- NEVER – Cannot be scheduled. This option can be used as a placeholder. The frequency must be changed to something else before the save can execute.
- EXPLICIT – When using scheduler type SCHEDULER, this option utilizes the string in EXPLICIT_INTERVAL to determine the scheduling frequency.
- CUSTOM – When using scheduler types INTERNAL or EXTERNAL, this option utilizes a user-defined schedule object, which has the frequency defined internally. Use this option if none of the above scheduling frequencies is appropriate.

/GROUPS=(group-name[,...])

/NOGROUPS

Specifies the names of groups of execution nodes on which this save can be run. Usually, only one group is specified. If neither the **/EXECUTION_NODES** nor **/GROUPS** qualifier is specified, then the node from which the command was issued is used as the node name.

/INCLUDE=(data-specification[,...])

/NOINCLUDE (Default)

The **/INCLUDE** qualifier allows you to enter a list of up to 24 data specifications to restore. The data specifications can be OpenVMS disks or files. However, all specifications must be of the same data select type, as specified in the **/DATA_SELECT_TYPE** qualifier. The include specification is stored in the default selection.

The include, exclude, and source node specifications, together with the data select type, are stored in a default selection object that has the name *save-name*_SAVE_SEL_DEF.

This selection object can be modified independently to add such refinements as exclude lists, and additional selections can be added to any save.

/NOINCLUDE removes the include file specification.

By default, there is no include specification.

/INCREMENTAL

/NOINCREMENTAL (Default)

The **/INCREMENTAL** qualifier specifies that the save should be an incremental save. If not specified, the save will be a full save or selective save based on the data-specification in the associated include specification or selections. **/NOINCREMENTAL** clears the incremental specification.

The default is **/NOINCREMENTAL**.

/INHERIT=save-name

This qualifier allows you to specify an existing save record from which the new object record inherits attribute values. MDMS supplies default values if you specify none. All attributes may be inherited with the exception of Save Name.

/KEEP

The **/KEEP** qualifier indicates that the save is not to be automatically deleted, and clears the delete interval.

/OWNER_NAME=node::username

/OWNER_NAME=group::username

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/PROLOGUE="string"

/NOPROLOGUE (Default)

The **/PROLOGUE** qualifier specifies a string to execute before each save thread (i.e. each include list item specification in the save). The string may be up to 80 characters and must be enclosed in quotes.

By default, no prologue command is executed.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/SELECTIONS=(selection[,...])

/NOSELECTIONS

When using the default selection, this qualifier is not needed, as MDMS adds it automatically. However, if you want to add your own selections to this save, you add them with the **/SELECTION** qualifier. By default a selection is added, but you can also remove selections with the **/REMOVE** qualifier, and replace selections with the **/REPLACE** qualifier. **/NOSELECTIONS** removes all selections from the save.

/SEQUENCE_OPTION=keyword

The **/SEQUENCE_OPTION** qualifier defines the type of sequencing used during the save operation. Select one of the following options:

- **SEQUENTIAL** – Performs each save thread sequentially. Each save thread must fully complete before the next one starts.
- **OVERLAPPED** – Starts the next save thread as soon as the backup drive or device is available for use.

The default is **SEQUENTIAL**.

/SKIP_TIME=delta-time
/NOSKIP_TIME (Default)

It is possible to run a save at any time, including times when it is not normally scheduled. You may, for example, want to run a save three hours early, or you may want to run an extra save for some reason.

By default, if you run a save at an unscheduled time, MDMS will also run it at its regularly scheduled time as well. If you do not want this, then enter a skip time when starting the save at the non-scheduled time. The skip time will be added to the start time, and MDMS will not schedule the save until at least after the skip_time value.

The default is **/NOSKIP_TIME**.

/START_DATE[=date]
/NOSTART_DATE

The **/START_DATE** qualifier is used to schedule a save to start at an unscheduled time. The **/BASE_DATE** is used to specify the normal scheduled times. By default, the start date is set to the base date if not separately specified.

The date value for the **/START_DATE** is optional. If not specified, the default is now. **/NOSTART_DATE** clears the next start date, and the save will not be scheduled.

Examples

The following commands creates a save **SAVE1**, which uses archive **SAVE1_ARC** and environment **SAVE1_ENV**. The data to save is **NOD001::USER1\$:[SMITH...]*.*;*** and **NOD001::USER2\$:[JONES...]*.*;***, which will be saved to their original locations. The selection type is **VMS_FILES**. The **/START** qualifier starts the save immediately, and the save is a **ONE_TIME_ONLY**:

```
$ MDMS CREATE SAVE SAVE1 -  
_ $ /ARCHIVE=SAVE1_ARC -  
_ $ /ENVIRONMENT=SAVE1_ENV -  
_ $ /INCLUDE=(USER1$:[SMITH...]*.*;*, -  
_ $ USER2$:[JONES...]*.*;*) -  
_ $ /EXECUTION_NODES=NOD001 -  
_ $ /START
```

The following command creates a save **SAVE2**, whose data selections are contained in user-created selection objects **SELECT1**, **SELECT2**, and **SELECT3**. This save is regularly scheduled with **DAILY_FULL_WEEKLY**, which means that a full save is performed on the base date and weekly anniversaries, and incremental saves are performed on all other days. The base time is 8:00 pm. The archive **SYSTEM_BACKUPS** and environment **SYSTEM_BACKUPS_ENV** are used:

```
$ MDMS CREATE SAVE SAVE2 -
```

```
_ $ /SELECTIONS=(SELECT1, SELECT2, SELECT3) -  
_ $ /ARCHIVE=SYSTEM_BACKUPS -  
_ $ /ENVIRONMENT=SYSTEM_BACKUPS_ENV -  
_ $ /FREQUENCY=DAILY_FULL_WEEKLY -  
_ $ /BASE_DATE=01-JAN-2001:20:00:00
```

MDMS CREATE SCHEDULE

MDMS CREATE SCHEDULE — Creates a new schedule definition in the MDMS database. Schedule is a new object type.

Format

```
MDMS CREATE SCHEDULE [schedule-name]
```

Equivalent ABS Command: None.

Parameters

schedule-name

Specifies the name of the schedule.

The maximum length of the schedule name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of schedule names may be entered.

Description

The **MDMS CREATE SCHEDULE** command creates a new schedule definition in the MDMS database. You can use schedules to execute any DCL command (including MDMS commands) at regular intervals.

Schedules are used by MDMS to execute saves and restores. However, these schedules are created implicitly by MDMS and do not need to be created with this command. However, you can modify CUSTOM schedules by using the **SET SCHEDULE** command.

When creating a new object record, MDMS supplies default values on attributes you do not specify. Alternatively, they can be inherited from a specified schedule using the **/INHERIT** qualifier.

Privileges

The request requires MDMS_CREATE_ALL.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/AFTER_SCHEDULE=(NAME=*schedule-name*, WHEN=*option*)
/NOAFTER_SCHEDULE

The **/AFTER_SCHEDULE** qualifiers specifies that this schedule is scheduled for execution after the AFTER_SCHEDULE name, under the circumstances defined in WHEN. This can be used to serialize a series of operations one after the other, and conditionally execute this schedule depending on the status of the AFTER_SCHEDULE.

The SCHEDULE should be a schedule name defined in the MDMS database. For WHEN, select one of the following:

ALL (Default)	Unconditionally execute this schedule when the AFTER_SCHEDULE completes
ERROR	Execute this schedule if the AFTER_SCHEDULE completed with ERROR or FATAL status

FATAL	Execute this schedule if the AFTER_SCHEDULE completed with FATAL status
NONE	Do not execute this schedule (can be used as a placeholder)
SUCCESS	Execute this schedule if the AFTER_SCHEDULE completed with SUCCESS or INFORMATIONAL status
WARNING	Execute this schedule if the AFTER_SCHEDULE completed with WARNING, ERROR or FATAL status

If an **/AFTER_SCHEDULE** name is specified, the default WHEN option is ALL.

/NOAFTER_SCHEDULE removes the dependency to execute after another schedule.

There is a restriction with using the **/AFTER_SCHEDULE** qualifier. Only those schedules (created automatically by MDMS) that have an associated save can be assigned to the **/AFTER_SCHEDULE** qualifier. Schedules that do NOT have an associated save cannot be assigned to the **/AFTER_SCHEDULE** qualifier. Hence, any schedule (one with an associated save, or one which executes DCL commands) can have a dependency on a schedule with an associated save, but not on a schedule which executes DCL commands. This is a current MDMS design limitation.

/COMMAND="string"

The **/COMMAND** qualifier specifies the command to submit to DCL when the schedule executes. Specify a valid DCL command line in quotes. This qualifier must be entered for the schedule to do any useful work.

/DATES=(date[,...])

/NODATES (Default)

The **/DATES** qualifier specifies on which days of the month you wish the schedule to execute. Valid values are 1-31. You can enter a date, a list of dates, a range of dates and a list of ranges (e.g. 1-7, 15-21). **/NODATES** indicates that all dates are valid (1-31) and other criteria are used for scheduling. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the dates list.

The default is **/NODATES**.

/DAYS=(day[,...])

/NODAYS (Default)

The **/DAYS** qualifier specifies on which days of the week you wish the schedule to execute. Valid values use at least the first three letters of the days of the week (English only). You can enter a day, a list of days, a range of days and a list of ranges (e.g. MON-WED, SAT-SUN). **/NODAYS** indicates that all day are valid (MON-SUN) and other criteria are used for scheduling. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the days list.

The default is **/NODAYS**.

/DESCRIPTION="text"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/EXCLUDE=(date[,...])

/NOEXCLUDE (Default)

The **/EXCLUDE** qualifier specifies specific dates that you wish to exclude from execution, even though the other scheduling parameters would normally schedule on that date. This can be used to exclude scheduled operations on holidays for example. Enter one or more dates in OpenVMS date format (04-JUL-2001), separated by commas. You can specify dates up to 9 years in the future. **/NOEXCLUDE** removes all dates from the exclude list. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the exclude list.

By default, no exclude dates are defined.

/INCLUDE=(date[,...])

/NOINCLUDE (Default)

The **/INCLUDE** qualifier specifies specific dates that you wish to include for execution, even though the other scheduling parameters would normally not schedule on that date. This can be used to include operations on special days that require special attention. Enter one or more dates in OpenVMS date format (31-DEC-2001), separated by commas. You can specify dates up to 9 years in the future. **/NOINCLUDE** removes all dates from the include list. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the include list.

By default, no include dates are defined.

/INHERIT=schedule-name

This qualifier allows you to specify an existing schedule record from which the new object record inherits attribute values. MDMS supplies default values if you specify none. All attributes may be inherited with the exception of Schedule Name.

/MONTHS=(month[,...])

/NOMONTHS (Default)

The **/MONTHS** qualifier specifies on which months of the year you wish the schedule to execute. Valid values use at least the first three letters of the months of the year (English only). You can enter a month, a list of months, a range of months and a list of ranges (e.g. JAN-MAR, JULSEP). **/NOMONTHS** indicates that all months are valid (JAN-DEC) and other criteria are used for scheduling. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the months list.

The default is **/NOMONTHS**.

/OWNER_NAME=node::username

/OWNER_NAME=group::username

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an

owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/TIMES=(time[,...])

/NOTIMES (Default)

The **/TIMES** qualifier specifies the times of the day you wish the schedule to execute. Enter from 1 to 100 times in OpenVMS format, separated by commas. For example (08:00, 10:00, 12:00, 14:00, 16:00). **/NOTIMES** clears out all times and effectively disables the schedule from execution. **/TIMES=00:00** signifies midnight.

By default, no times are set.

/WEEKS=(week[,...])

/NOWEEKS (Default)

The **/WEEKS** qualifier specifies on which weeks of the year you wish the schedule to execute. Valid values are 1-53. You can enter a week, a list of weeks, a range of weeks and a list of ranges (e.g. 1-7, 15-21, 31-53). **/NOWEEKS** indicates that all weeks are valid (1-53) and other criteria are used for scheduling. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the weeks list.

The default is **/NOWEEKS**.

Examples

The following command creates a schedule called **SCHED1**, that executes every Sunday at 20:00 by issuing the command procedure **@USER1:[SMITH]SCHED1.COM**:

```
$ MDMS CREATE SCHEDULE SCHED1 -  
_ $ /DAYS=SUN -  
_ $ /TIMES=20:00 -  
_ $ /COMMAND="@USER1:[SMITH]SCHED1.COM"
```

The following command creates a schedule called **SCHED2**, that executes every three months on the first of January, April, July, and October at midnight, and executes an MDMS command to delete saves:

```
$ MDMS CREATE SCHEDULE SCHED2 -  
_ $ /MONTHS=(JAN, APR, JUL, OCT) -  
_ $ /DATES=1  
_ $ /TIMES=00:00
```

```
_$ /COMMAND="MDMS DELETE SAVE *QUARTER*"
```

MDMS CREATE SELECTION

MDMS CREATE SELECTION — Creates a new selection definition in the MDMS database. Selection is a new object type.

Format

```
MDMS CREATE SELECTION [selection-name]
```

Equivalent ABS Command: None.

Parameters

selection-name

Specifies the name of the selection.

The maximum length of the selection name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of selection names may be entered.

Description

The **MDMS CREATE SELECTION** command creates a new selection definition in the MDMS database. Selections are used to specify the data to be saved or restored in those objects. Under most circumstances, you can specify the data to be saved or restored in the appropriate save or restore object, and MDMS will implicitly create a default selection. However, there may be times that you wish to create your own selections, and add them to the relevant save or restore object. This may be because you wish to specify a mixture of data select types in a save/restore, or specific file dates in a save.

You may modify both user-created and default selections using the **SET SELECTION** command.

Many of the qualifiers for selections map to the positional qualifiers on saves and restores with the pre-V4 ABS **SAVE** and **RESTORE** commands.

When creating a new object record, MDMS supplies default values on attributes you do not specify. Alternatively, they can be inherited from a specified SELECTION using the **/INHERIT** qualifier.

Privileges

The request requires MDMS_CREATE_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/AGENT_QUALIFIERS="*string*"

/NOAGENT_QUALIFIERS (Default)

The **/AGENT_QUALIFIERS** qualifier allows you to specify a backup-agent specific qualifier or set of qualifiers. This qualifier is valid for all data select types. Specify up to 80 characters enclosed in quotes. **/NOAGENT_QUALIFIERS** removes all the agent qualifiers.

By default, no agent qualifiers are defined.

/BEFORE_DATE=*date*

/NOBEFORE_DATE (Default)

The **/BEFORE_DATE** is used to select files that have a date before the specified date. The type of date in question is specified by the **/DATE_TYPE** qualifier. If **/NOBEFORE_DATE** is specified, files are not selected using a before date.

/BEFORE_DATE is applicable only to saves.

The default is that no before date is specified.

/CONFLICT_OPTIONS=option

The **/CONFLICT_OPTIONS** is used to determine the disposition of a file if an existing file already exists. This qualifier is applicable only to restores. Select one of the following:

- **NEW_VERSION** – Create a new version of the file
- **OVERLAY_VERSION** – Overlay the data in the current version of the file, but maintain the existing file header
- **REPLACE_VERSION** – Delete the existing version and replace it with the archive version
- **RETAIN_VERSION** (Default) – Leave the existing version and do not restore the version from the archive

If not specified, the default is **RETAIN_VERSION**.

/DATA_SELECT_TYPE=keyword

The **/DATA_SELECT_TYPE** qualifier specifies the type of data to restore in the default selection.

Select one of the following options:

- **VMS_FILES**
- **VMS_SAVESET**

If not specified, MDMS attempts to determine the data select type based on the include file syntax. If MDMS cannot determine this, **VMS_FILES** is used.

/DATE_TYPE

The **/DATE_TYPE** qualifier specifies which date in a file is used using the **/BEFORE_DATE** and/or **/SINCE_DATE** in a selection. Select one of the following date types:

- **BACKUP** – The BACKUP date is used for selection.
- **EXPIRED** – The expiration date is used for selection. This can also be configured as last access date.
- **CREATED** – The creation date is used for selection.
- **MODIFIED** (Default) – The revision date is used for selection.

If not specified, **MODIFIED** is the default.

/DESCRIPTION="text"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/EXCLUDE=(*data-specification*[,...])

/NOEXCLUDE (Default)

The **/EXCLUDE** qualifier is used to exclude files that otherwise would be included with the **/INCLUDE** qualifier. This is used to exclude files for save or restore when a more general wildcard specification is specified for the include.

By default, no files are excluded.

/INCLUDE=(*data-specification*[,...])

/NOINCLUDE

The **/INCLUDE** qualifier allows you to enter a list of up to 24 data specifications to restore. The data specifications can be OpenVMS disks or files. However, all specifications must be of the same data select type, as specified in the **/DATA_SELECT_TYPE** qualifier.

/NOINCLUDE clears the include specification.

See [MDMS CREATE RESTORE](#) for the note on maintaining file directory structure during the Restore operation.

/INHERIT=*selection-name*

This qualifier allows you to specify an existing selection record from which the new object record inherits attribute values. MDMS supplies default values if you specify none. All attributes may be inherited with the exception of Selection Name.

/OWNER_NAME=*node::username*

/OWNER_NAME=*group::username*

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/SINCE_DATE=*date*
/NOSINCE_DATE

The **/SINCE_DATE** is used to select files that have a date after the specified date. The type of date in question is specified by the **/DATE_TYPE** qualifier. If **/NOSINCE_DATE** is specified, files are not selected using a since date.

You can specify date as an absolute date, as a combination of absolute and delta times, or as one of the following keywords: TODAY (default), TOMORROW, or YESTERDAY.

/SINCE_DATE is applicable only to saves.

The default is that no since date is specified.

Examples

The following command creates a selection SEL1 that includes files USER1:[BACON]*.*;* that have been modified since 24-May-2001, but excluding all .LOG and .TMP files in that directory:

```
$ MDMS CREATE SELECTION SEL1 -  
_ $ /INCLUDE=(USER1:[BACON]*.*;* ) -  
_ $ /EXCLUDE=(USER1:[BACON]*.LOG;* , *.TMP;* ) -  
_ $ /SINCE=24-MAY-2001 -  
_ $ /DATE_TYPE=MODIFIED -
```

The following command creates a selection SEL2 that includes files /usr/smith/ on node XTRIX:

```
$ MDMS CREATE SELECTION SEL2 -  
_ $ /INCLUDE=(/usr/smith/) -  
_ $ /SOURCE_NODES=XTRIX
```

MDMS CREATE VOLUME

MDMS CREATE VOLUME — Creates a new volume definition in the MDMS volume database.

Format

MDMS CREATE VOLUME [*volume-id* [, ...]]

Equivalent STORAGE Command: **STORAGE ADD VOLUME**

Parameters

volume-id

Specifies the volume ID of the volume to be added. The volume ID is the external label for the volume. Alternatively, a volume range (separated by a dash) may be specified. A volume range is a numeric range for up to the last five characters of the volume ID. Example ranges are (ABC001-ABC250) and (T20120-T20125).

The maximum length of the volume ID is 6 characters. Valid characters are " , ! , " , % - ? , A - Z , _ " .

There is a limit of 1000 volumes in a valid range, and ranges must be numerically increasing.

Either a volume ID or a volume range must be entered in the command, unless the **/VISION** qualifier is used to automatically determine volume identifiers in a jukebox.

A comma-separated list of volume IDs or ranges may be entered.

Description

The **MDMS CREATE VOLUME** command creates a new volume definition in the MDMS volume database. The external label will be used to track the volume in the database. The external label must match the on-tape internal volume label when the volume is initialized.

Privileges

The request requires MDMS_CREATE_ALL, MDMS_CREATE_VOLUME, or MDMS_CREATE_POOL.

If the user is creating a volume in a named pool to which they are authorized, MDMS_CREATE_POOL is sufficient. The **/POOL** qualifier must be specified. Otherwise the request requires MDMS_CREATE_VOLUME or MDMS_CREATE_ALL.

Several protected qualifiers also require MDMS_SET_PROTECTED since these attributes are normally managed by MDMS and should not be modified by the user, unless in case of recovery from abnormal situations.

Restrictions

The **/VISION** qualifier is incompatible with the *volume-id* parameter.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object

- **CONTROL** – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ACCOUNT="text"

Defines the account name of the volume. The account name may be up to 31 characters. If it contains spaces, it must be enclosed in quotation marks. Specify "" to clear the account name.

This qualifier requires the right MDMS_SET_PROTECTED.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/ALLOCATED_DATE=date

/NOALLOCATED_DATE

Specifies the date the volume was allocated. Normally this is set by MDMS. Specify a VMS absolute date and time. Specify **/NOALLOCATED_DATE** to clear the allocated date.

This qualifier requires the right MDMS_SET_PROTECTED.

/AVAILABLE

The **/AVAILABLE** qualifier moves a volume from the UNAVAILABLE state to the state it was previously in prior to the UNAVAILABLE state. The volume may then be moved into the TRANSITION or FREE state if the scratch date and/or transition time have expired.

/BLOCK_FACTOR=number

Specifies the block factor for the volume. The default is a block factor of zero.

/BRAND="text"

The media manufacturer. The maximum length of the brand name is 31 characters. If it contains spaces it must be contained in quotation marks. Specify "" to clear the brand.

/CLEANED_DATE=date

/NOCLEANED_DATE

This qualifier specifies the date the volume was last cleaned and is entered as a VMS absolute time. Specify **/NOCLEANED_DATE** to clear the cleaned date. The default cleaned date is the date/time the volume was created.

/CREATION_DATE=*date*
/NOCREATION_DATE

The date the volume is created. This attribute is set by MDMS, but may be overridden if necessary. The default creation date is the date/time the volume was created. This qualifier requires the right MDMS_SET_PROTECTED.

/DEALLOCATED_DATE=*date*
/NODEALLOCATED_DATE

This qualifier specifies the actual deallocation date for the volume. Specify a VMS absolute time. This date is normally set by MDMS.

This qualifier requires the right MDMS_SET_PROTECTED.

/DESCRIPTION=*"text"*

Comments about the volume. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. Specify "" to clear the description.

/DRIVE=*drive-name*
/NODRIVE

This is a protected field that should be modified only to recover on error. Use the **LOAD** and **UNLOAD** commands to manipulate the drive name under normal operation. This qualifier specifies the drive that the volume currently resides in or last resided in. This is normally set up by MDMS. To clear the drive, specify **/NODRIVE**.

This qualifier requires the right MDMS_SET_PROTECTED.

/FORMAT=*keyword*

Specifies the format of the tape. Possible values are as follows:

ASCII
BACKUP
EBCDIC
NONE (Default)
RMUBACKUP

/FREED_DATE=*date*
/NOFREED_DATE

Specifies the date the volume was last freed (i.e. put in the FREE state). Specify an OpenVMS absolute date and time. This is normally set up by MDMS. To clear the freed date, specify **/NOFREED_DATE**.

This qualifier requires the right MDMS_SET_PROTECTED.

/INHERIT=*volume-id*

This qualifier allows you to specify an existing volume record to inherit default attributes. The default is that MDMS supplies values you do not supply on creation. All attributes may be inherited with the exception of the following protected fields:

Account

Allocate date

Accessed date

Available state	Creation date	Deallocated date
Drive name	Freed date	Initialized date
Job name	Jukebox name	Magazine name
Owner	Placement	Slot
State	User name	Next and Previous Volumes (not settable)

/INITIALIZED_DATE=date

/NOINITIALIZED_DATE

Specifies the date the volume was last initialized. Specify a VMS absolute date and time. This is normally set up by MDMS. To clear the initialized date, specify ***/NOINITIALIZED_DATE***.

This qualifier requires the right MDMS_SET_PROTECTED.

/IO_ERROR_COUNT=number

This qualifier allows you to set the number of I/O errors on the volume. The default value is zero.

/JOB_NAME="text"

This qualifier allows you to specify the last job that accessed the volume. The job name can be from 0 to 31 characters. If it contains spaces, it must be enclosed in quotation marks. Specify "" to clear the job name.

This qualifier requires the right MDMS_SET_PROTECTED.

/JUKEBOX=jukebox-name

/NOJUKEBOX

This is a protected field that should be modified only to recover on error. Use the **MOVE VOLUME** command to manipulate the jukebox name under normal operation. This qualifier allows you to specify that the volume is currently residing or last resided in the specified jukebox.

The maximum length of a jukebox name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

This is normally set up by MDMS. To clear the jukebox name, specify ***/NOJUKEBOX***.

This qualifier requires the right MDMS_SET_PROTECTED.

/LAST_ACCESS_DATE=date

/NOLAST_ACCESS_DATE

Specifies the date the volume was last loaded by MDMS. Specify a VMS absolute date and time. This is normally set up by MDMS. To clear the last access date, specify ***/NOLAST_ACCESS_DATE***.

This qualifier requires the right MDMS_SET_PROTECTED.

/MAGAZINE=magazine-name

/NOMAGAZINE

This is a protected field that should be modified only to recover on error. Use the **MOVE VOLUME** command to manipulate the magazine name under normal operation. This qualifier specifies the

magazine name if the volume resides in a magazine. This is normally set up by MDMS. To clear the magazine name, specify **/NOMAGAZINE**.

This qualifier requires the right MDMS_SET_PROTECTED.

/MEDIA_TYPES=(*media-type*[,...])
/NOMEDIA_TYPES

The media type qualifier allows you to add the media types that the volume can support. Multiple media types are supported prior to the volume being initialized. After initialization, a volume can only support one media type. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace objects in the list, rather than adding them by default. To specify the volume supports no media types, enter **/NOMEDIA_TYPES**. If a volume is created with no media types, the default media type from the domain record is used.

/MOUNT_COUNT=*number*

Specifies the number of times the volume has been loaded by MDMS. Normally set up by MDMS. The default mount count is zero.

/OFFSITE=([LOCATION=*location*]**,**[NO]DATE[=*date*]**)**
/NOOFFSITE

This qualifier specifies the date that the volume is to be taken off site and the off-site location. The location field is required when using the **/OFFSITE** qualifier if no location has been previously specified. The LOCATION keyword cannot be negated and if specified must point to a valid location object. Specify a VMS absolute or delta time in the date field. The NODATE keyword may be used to remove the date. This has the effect of disabling the schedule for the volume, while retaining the location. To clear the off-site date specify **/NOOFFSITE**. If a volume is under magazine control, the **/OFFSITE** qualifier is not allowed. The volume uses the values in the magazine.

The default off-site location is that stored in the domain object.

/ONSITE=([LOCATION=*location*]**, **[NO]DATE=*date***)**
/NOONSITE

This qualifier specifies the date that the volume is to be brought back on site and the on-site location. The location field is required when using the **/ONSITE** qualifier if no location has been previously specified. The LOCATION keyword cannot be negated and if specified must point to a valid location object. Specify a VMS absolute or delta time in the date field. The NODATE keyword may be used to remove the date. This has the effect of disabling the schedule for the volume, while retaining the location. To clear the on-site date specify **/NOONSITE**. If a volume is under magazine control, the **/ONSITE** qualifier is not allowed. The volume uses the values in the magazine.

The default on-site location is that stored in the domain object.

/OWNER_NAME=*node::username*
/OWNER_NAME=*group::username*
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/OWNER=[owner-name]
/NOOWNER

This qualifier specifies the owner UIC of a volume. The owner UIC field must be a UIC in the format [USER] or [group, user]. This is normally set up by MDMS on allocate volume. To clear the owner UIC field, specify **/NOOWNER_NAME_UIC**.

This qualifier requires the right MDMS_SET_PROTECTED.

/PLACEMENT=keyword

This is a protected field that should be modified only to recover on error. Use the **MOVE**, **LOAD**, or **UNLOAD** commands to manipulate the placement field under normal operation. This qualifier defines the current placement of the volume. This is normally managed by MDMS. Select one of the following options:

DRIVE
ONSITE
MAGAZINE
JUKEBOX
OFFSITE
MOVING

If a magazine name is specified on the **/MAGAZINE** qualifier, the volume placement can be in one of three states: MAGAZINE, DRIVE, or MOVING.

During a **MOVE**, **LOAD**, or **UNLOAD** operation, a volume's placement may be set to MOVING, indicating that the volume is being moved. If a volume is in a magazine, it is set to MOVING when the volume is being loaded or unloaded to/from a drive.

This qualifier requires the right MDMS_SET_PROTECTED.

/POOL=pool-name
/NOPOOL (Default)

The pool in which the volume belongs. The maximum length of the pool name is 31 characters. Spaces are not allowed in the pool name. If no pool is specified, the volume is considered to be part of a scratch pool and can be allocated by any user.

/PREINITIALIZED
/NOPREINITIALIZED (Default)

This qualifier specifies whether the volume has been initialized before creation. If **/PREINITIALIZED** is specified, the volume is placed in the FREE state rather than the UNINITIALIZED state.

/PROTECTION=*protection*

The protection code for the volume. Use the standard OpenVMS protection code format. This protection is written to volumes when initialized by MDMS. If not specified, the default protection from the domain record is used.

/PURCHASED_DATE=*date***/NOPURCHASED_DATE**

The date when the volume was purchased. The date should be specified as an OpenVMS absolute time. Specify **/NOPURCHASED_DATE** to clear the purchased date. The default purchased date is the date/time that the volume was created.

/RECLength=*number*

This qualifier specifies the record length used on the volume. The default record length is zero.

/RELEASE

This qualifier puts the volume into the FREE state from the TRANSITION state. It is not valid to release an allocated volume.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/RETAIN

This qualifier puts the volume that is in the FREE state or TRANSITION state back into the allocated state with the former owner as the current owner.

/SCRATCH_DATE=*date***/NOSCRATCH_DATE**

The **/SCRATCH_DATE** qualifier specifies the planned date to return the volume from the ALLOCATED state to either the TRANSITION state or the FREE state. Specify **/NOSCRATCH_DATE** if the volume should not automatically transition from the ALLOCATED state.

/SLOTS=(*range[,...]*)**/NOSLOTS**

This qualifier specifies the jukebox or magazine slot that the volume currently resides in. Specify a number in the range of the jukebox or magazine. This is normally set up by MDMS. If the volume does not currently reside in a jukebox or magazine slot, specify **/NOSLOTS**.

If no volume ID is specified, or a volume range is specified, you can specify a slot range, and each volume will be placed in each slot in order. For a single volume, specify a single slot. This qualifier requires the right MDMS_SET_PROTECTED.

/SPACES=(*range[,...]*)
/NOSPACES

This qualifier specifies the non-jukebox space in the specified location that the volume resides in. If the volume does not reside in a location space, specify **/NOSPACES**. Use a space range only when creating multiple volumes—each volume will be placed in each space in order. For a single volume, specify a single space.

/STATE=*keyword*

This is a protected field that should be modified only to recover on error. Use the **ALLOCATE VOLUME** or **DEALLOCATE VOLUME** command to manipulate the state field under normal operation. This qualifier allows you to modify the state of the volume. This is normally set up by MDMS and manual modification is not recommended. The keyword values are:

ALLOCATED
FREE
TRANSITION
UNAVAILABLE
UNINITIALIZED

This qualifier requires the right MDMS_SET_PROTECTED.

/TIMES_CLEANED=*number*

This qualifier allows you to specify the number of times the volume has been cleaned. The default is zero.

/TRANSITION_TIME=*delta-time*
/NOTRANSITION_TIME

The **/TRANSITION_TIME** qualifier specifies that the volume enters the TRANSITION state when the scratch date is reached, and is to stay in the TRANSITION state for the specified delta time. When the transition time has expired, the volume enters the FREE state. The **/NOTRANSITION_TIME** qualifier specifies that the volume enters the FREE state directly at the scratch date.

/UNAVAILABLE

Puts the volume in the UNAVAILABLE state. The previous state is retained for when the volume is made available again.

/USER_NAME=*username*
/NOUSER_NAME

Specifies the user for the volume. The username can be from 1-31 characters, and must reflect an authorized VMS username. To clear the username, enter **/NOUSER_NAME**.

This qualifier requires the right MDMS_SET_PROTECTED.

/VISION

This qualifier specifies that volume IDs are to be read using a vision system in a suitably equipped jukebox. No volume ID or volume range should be specified when using this qualifier. Valid only for MRD jukeboxes equipped with a VISION system.

Examples

The following command creates a new volume definition for volume ABC001, of media type TK85K, which is stored in the location SHELF2, space 1:

```
$ MDMS CREATE VOLUME ABC001 /MEDIA=TK85K -  
/ONSITE=(LOCATION=SHELF2) /SPACES=1
```

The following command creates volume records for the newly imported volumes in slots 0–50 of the jukebox JUKE_1, and the volume labels are generated using the vision system, and other attributes are inherited from volume DEF000:

```
$ MDMS CREATE VOLUME /INHERIT=DEF000 -  
/JUKEBOX=JUKE_1 /SLOTS=(0-50) /VISION
```

The following command creates seven volume records HS0001–HS0007, which are stored in magazine TX877A in slots 0–6 respectively, and other attributes are inherited from volume TK85MG:

```
$ MDMS CREATE VOLUME HS0001-HS0007 /INHERIT=TK85MG -  
/MAGAZINE=TX877A /SLOTS=(0-6)
```

MDMS DEALLOCATE DRIVE

MDMS DEALLOCATE DRIVE — Deallocates a drive.

Format

```
MDMS DEALLOCATE DRIVE [drive-name]
```

Equivalent STORAGE Command: None.

Parameters

drive-name

Specifies the drive name to be deallocated. Specify a drive name or the logical name previously defined in the allocate.

The maximum length of the drive name or logical name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

Description

The **MDMS DEALLOCATE DRIVE** command deallocates a drive. If a logical name is supplied for the *drive-name* parameter, the logical name will be deassigned.

Privileges

The request requires MDMS_DEALLOCATE_ALL or MDMS_DEALLOCATE_OWN.

Restrictions

Can only be issued by the process that allocated the drive. If that process terminates, the drive is automatically deallocated.

Qualifiers

None.

Examples

The following command deallocates the drive assigned to the MYDRIVE logical and deassigns the logical name MYDRIVE:

```
$ MDMS DEALLOCATE DRIVE MYDRIVE
```

The following command deallocates drive \$1\$MUA1:

```
$ MDMS DEALLOCATE DRIVE $1$MUA1
```

MDMS DEALLOCATE VOLUME

MDMS DEALLOCATE VOLUME — Puts an allocated volume in either the transition state or the free state, depending on volume attributes that can be overridden in the command.

Format

```
MDMS DEALLOCATE VOLUME [volume-id]
```

Equivalent STORAGE Command: **STORAGE DEALLOCATE**

Parameters

volume-id

Specifies the volume ID of the volume to deallocate. This parameter is required except when the **/SCHEDULE** qualifier is given, in which case it must be omitted.

The maximum length of a volume ID is 6 characters. Valid characters are " ,! , % - ? , A - Z , _ ".

Description

The **MDMS DEALLOCATE VOLUME** command puts an allocated volume in either the TRANSITION state or the FREE state, depending on volume attributes that can be overridden in the command.

In addition, this command will update the state of the volume to FREE if the deallocated date plus transition time has expired, or if you issue a **DEALLOCATE VOLUME/STATE=FREE** command. If you deallocate a volume set, the volume set relationships are invalidated, but can be recovered with a **SET VOLUME/REATIN** command. The volume set relationships are deleted on the next allocation of the volume.

Privileges

The request requires MDMS_DEALLOCATE_ALL or MDMS_DEALLOCATE_OWN.

If the user is deallocating a volume owned by them, MDMS_DEALLOCATE_OWN is sufficient. If the user is deallocating on behalf of another user by using the **/USER_NAME** qualifier, then the request requires MDMS_DEALLOCATE_ALL.

If the **/STATE** or **/TRANSITION_TIME** qualifiers are entered, the user needs MDMS_SET_ALL if the **/USER_NAME** qualifier is specified, or else MDMS_SET_OWN or MDMS_SET_POOL.

Restrictions

The **/SCHEDULE** qualifier and *volume-id* parameter are incompatible.

The **/SCHEDULE** and **/NOVOLSET** qualifiers are incompatible.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/SCHEDULE

The **/SCHEDULE** qualifier selects volumes whose scratch dates or freed dates have expired, and moves them into the TRANSITION state or FREE state respectively. No *volume-id* parameter is allowed when this qualifier is given.

/STATE=state

Specifies the state in which to place the deallocated volume. The valid state values are FREE and TRANSITION. If the state is not specified and no transition time is specified, the deallocation state in the domain record is used.

This qualifier requires the right MDMS_SET_OWN or MDMS_SET_POOL, or MDMS_SET_ALL if the **/USER_NAME** qualifier is used.

/TRANSITION_TIME=delta-time

/NOTRANSITION_TIME

Modifies the amount of time, as a delta time, that a volume will remain in the TRANSITION state before moving to the FREE state. Use the standard OpenVMS delta time format to specify a delta time for the transition duration. If not specified, the existing transition time in the volume record is used, and if none, the default transition time in the domain record is used.

This qualifier requires the right MDMS_SET_OWN or MDMS_SET_POOL, or MDMS_SET_ALL if the **/USER_NAME** qualifier is used.

/USER_NAME=username

Deallocate a volume which is owned by the specified user. The maximum length of the username is 31 characters. This qualifier requires the right MDMS_DEALLOCATE_ALL.

/VOLSET (Default)

/NOVOLSET

Indicates that the entire volume set which contains the volume ID is to be deallocated. The default of **/VOLSET** deallocates all volumes in the volume set. If **/NOVOLSET** is specified, the specified

volume is deallocated. This qualifier is ignored if the volume is not in a set. You cannot deallocate the first volume in a volume set using **/NOVOLSET**.

Examples

The following command deallocates volume VOL008 for the current user and places the volume in the FREE state:

```
$ MDMS DEALLOCATE VOLUME VOL008 /NOVOLSET /STATE=FREE
```

The following command deallocates volume ACF342 that was allocated to user SMITH:

```
$ MDMS DEALLOCATE VOLUME ACF342 /USER_NAME=SMITH
```

The following command deallocates all volumes in the set that contains volume VOL002 for the current user. The volumes are also unbound from the volume set:

```
$ MDMS DEALLOCATE VOLUME VOL002 /VOLSET
```

A volume set contains the volumes VOL1, VOL2, VOL3, VOL4, and VOL5. The following command deallocates volume VOL2 for the current user and leaves two volume sets (VOL1, VOL2 and VOL4, VOL5):

```
$ MDMS DEALLOCATE VOLUME VOL3 /NOVOLSET
```

The following command deallocates all volumes whose scratch date has expired. It also moves volumes into the FREE state if the freed date has expired:

```
$ MDMS DEALLOCATE VOLUME/SCHEDULE
```

MDMS DELETE ARCHIVE

MDMS DELETE ARCHIVE — Deletes a specified archive definition from the MDMS database.

Format

```
MDMS DELETE ARCHIVE [archive-name] [, ...]
```

Equivalent ABS Command: **DELETE STORAGE_CLASS**

Parameters

archive-name

Specifies the name of the archive.

The maximum length of the archive name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of archive names may be entered.

Description

The **MDMS DELETE ARCHIVE** command deletes a specified archive definition from the MDMS database.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/OVERRIDE

The **/OVERRIDE** qualifier allows you to delete the archive even with outstanding references to it. This is not recommended unless you are sure that the archive will no longer be needed.

Examples

The following command deletes the archive definition for SYSTEM_BACKUPS_SPEC:

```
$ MDMS DELETE ARCHIVE SYSTEM_BACKUPS_SPEC
```

MDMS DELETE CATALOG

MDMS DELETE CATALOG — Deletes a specified catalog definition from the local ABS database.

Format

```
MDMS DELETE CATALOG [catalog-name] [, ...]
```

Equivalent ABS Command: **RUN SYS\$SYSTEM:ABS\$CATALOG_OBJECT**

Parameters

catalog-name

Specifies the name of the catalog.

The maximum length of the catalog name is 26 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of catalog names may be entered.

Description

The **MDMS DELETE CATALOG** command deletes a specified catalog definition from the local ABS database and deletes the related catalog files.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/NODE=*node-name*

The **/NODE** qualifier specifies on which node the catalog will be deleted.

The default is the node on which the **DELETE CATALOG** command is entered.

/OVERRIDE

The **/OVERRIDE** qualifier allows you to delete the catalog even with outstanding references to it. This is not recommended unless you are sure that the catalog will no longer be needed.

Examples

The following command deletes the catalog definition for `SYSTEM_CATALOG`. It performs the delete even if there are references to this catalog:

```
$ MDMS DELETE CATALOG SYSTEM_CATALOG/OVERRIDE
```

MDMS DELETE DRIVE

MDMS DELETE DRIVE — Deletes a specified drive definition from the MDMS database. The drive must be deallocated prior to being deleted.

Format

```
MDMS DELETE DRIVE [drive-name] [, ...]
```

Equivalent **STORAGE** Command: None.

Parameters

drive-name

Specifies the name of the drive.

The maximum length of the drive name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of drive names may be entered.

Description

The **MDMS DELETE DRIVE** command deletes a specified drive definition from the MDMS database.

Privileges

The request requires `MDMS_DELETE_ALL`.

Restrictions

None.

Qualifiers

None.

Examples

The following command deletes the drive definition for \$1\$MUA5:

```
$ MDMS DELETE DRIVE $1$MUA5
```

MDMS DELETE ENVIRONMENT

MDMS DELETE ENVIRONMENT — Deletes a specified environment definition from the MDMS database.

Format

```
MDMS DELETE ENVIRONMENT [environment-name] [, ...]
```

Equivalent ABS Command: **DELETE ENVIRONMENT**

Parameters

environment-name

Specifies the name of the environment.

The maximum length of the environment name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of environment names may be entered.

Description

The **MDMS DELETE ENVIRONMENT** command deletes a specified environment definition from the MDMS database.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/OVERRIDE

The **/OVERRIDE** qualifier allows you to delete the environment even with outstanding references to it. This is not recommended unless you are sure that the environment will no longer be needed.

Examples

The following command deletes the `ENVIRONMENT` definition for `KELLIE_ENV`:

```
$ MDMS DELETE ENVIRONMENT KELLIE_ENV
```

MDMS DELETE GROUP

MDMS DELETE GROUP — Deletes a specified group definition from the MDMS database.

Format

```
MDMS DELETE GROUP [group-name] [, ...]
```

Equivalent `STORAGE` Command: None.

Parameters

group-name

Specifies the name of the group.

The maximum length of the group name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of group names may be entered.

Description

The **MDMS DELETE GROUP** command deletes a specified group definition from the MDMS database.

Privileges

The request requires `MDMS_DELETE_ALL`.

Restrictions

None.

Qualifiers

None.

Examples

The following command deletes the group definition for HOSER:

```
$ MDMS DELETE GROUP HOSER
```

MDMS DELETE JUKEBOX

MDMS DELETE JUKEBOX — Deletes a specified jukebox definition from the MDMS database. Before deleting a jukebox, it is highly recommended that any drives defined as being in the jukebox are also deleted (or modified), and volumes and magazines are moved out of the jukebox.

Format

```
MDMS DELETE JUKEBOX [jukebox-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

jukebox-name

Specifies the name of the jukebox.

The maximum length of the jukebox name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of jukebox names may be entered.

Description

The **MDMS DELETE JUKEBOX** command deletes a specified jukebox definition from the MDMS database. Before deleting a jukebox, it is highly recommended that any drives defined as being in the jukebox are also deleted (or modified), and volumes and magazines are moved out of the jukebox.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

None.

Qualifiers

None.

Examples

The following command deletes the definition for jukebox JUKE_1:

```
$ MDMS DELETE JUKEBOX JUKE_1
```

MDMS DELETE LOCATION

MDMS DELETE LOCATION — Deletes a specified location definition from the MDMS database.

Format

```
MDMS DELETE LOCATION [location] [,...]
```

Equivalent STORAGE Command: None.

Parameters

location

Specifies the name of the location.

The maximum length of the location is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of locations may be entered.

Description

The **MDMS DELETE LOCATION** command deletes a specified location definition from the MDMS database.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

None.

Qualifiers

None.

Examples

The following command deletes the location definition for ROOM_1:

```
$ MDMS DELETE LOCATION ROOM_1
```

MDMS DELETE MAGAZINE

MDMS DELETE MAGAZINE — Deletes a specified magazine from the MDMS database.

Format

```
MDMS DELETE MAGAZINE [magazine-name] [, ...]
```

Equivalent STORAGE Command: **STORAGE REMOVE MAGAZINE**

Parameters

magazine-name

Specifies the name of the magazine.

The maximum length of the magazine name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of magazine names may be entered.

Description

The **MDMS DELETE MAGAZINE** command deletes a specified magazine definition from the MDMS database.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

None.

Qualifiers

None.

Examples

The following command deletes the magazine MAG001:

```
$ MDMS DELETE MAGAZINE MAG001
```

MDMS DELETE MEDIA_TYPE

MDMS DELETE MEDIA_TYPE — Deletes a specified media type definition from the MDMS database.

Format

```
MDMS DELETE MEDIA_TYPE [media-type] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

media-type

Specifies the name of the media type.

The maximum length of the media type name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of media types may be entered.

Description

The **MDMS DELETE MEDIA_TYPE** command deletes a specified media type definition from the MDMS database.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

None.

Qualifiers

None.

Examples

The following command deletes the definition for media type TK87K:

```
$ MDMS DELETE MEDIA_TYPE TK87K
```

MDMS DELETE NODE

MDMS DELETE NODE — Deletes a specified node definition from the MDMS database.

Format

```
MDMS DELETE NODE [node-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

node-name

Specifies the name of the node.

The maximum length of the node name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of node names may be entered.

Description

The **MDMS DELETE NODE** command deletes a specified node definition from the MDMS database.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

None.

Qualifiers

None.

Examples

The following command deletes the node name FUDGE:

```
$ MDMS DELETE NODE FUDGE
```

MDMS DELETE POOL

MDMS DELETE POOL — Deletes a specified pool definition from the MDMS database.

Format

```
MDMS DELETE POOL [pool-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

pool-name

Specifies the name of the volume pool.

The maximum length of the pool name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of pool names may be entered.

Description

The **MDMS DELETE POOL** command deletes a specified pool definition from the MDMS database.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

None.

Qualifiers

None.

Examples

The following command deletes pool SLSPool:

```
$ MDMS DELETE POOL SLSPool
```

MDMS DELETE RESTORE

MDMS DELETE RESTORE — Deletes a specified restore definition from the MDMS database.

Format

```
MDMS DELETE RESTORE [restore-name] [, ...]
```

Equivalent STORAGE Command: **DELETE RESTORE**

Parameters

restore-name

Specifies the name of the restore.

The maximum length of the restore name is 39 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of restore names may be entered. If **/SCHEDULE** is entered, a wildcard restore name may be specified.

Description

The **MDMS DELETE RESTORE** command deletes a specified restore definition from the MDMS database.

Privileges

The request requires MDMS_DELETE_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/OVERRIDE

The **/OVERRIDE** qualifier allows you to delete the restore even with outstanding references to it. This is not recommended unless you are sure that the restore will no longer be needed.

/SCHEDULE

Deletes the restore request only if its frequency is marked as **ONE_TIME_ONLY** and its delete interval has expired. Usable with a wildcard restore name.

Examples

The following command deletes the restore definition for **BOBS_FILES_REST**:

```
$ MDMS DELETE RESTORE BOBS_FILES_REST
```

MDMS DELETE SAVE

MDMS DELETE SAVE — Deletes a specified save definition from the MDMS database.

Format

```
MDMS DELETE SAVE [save-name] [, ...]
```

Equivalent STORAGE Command: **DELETE SAVE**

Parameters

save-name

Specifies the name of the save.

The maximum length of the save name is 39 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of save names may be entered. If **/SCHEDULE** is entered, a wildcard save name may be specified.

Description

The **MDMS DELETE SAVE** command deletes a specified save definition from the MDMS database.

Privileges

The request requires **MDMS_DELETE_ALL**.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/OVERRIDE

The **/OVERRIDE** qualifier allows you to delete the save even with outstanding references to it. This is not recommended unless you are sure that the save will no longer be needed.

/SCHEDULE

Deletes the save request only if its frequency is marked as **ONE_TIME_ONLY**, and its delete interval has expired. Usable with a wildcard save name.

Examples

The following command deletes the **SAVE** definition for **DENNIS_SAVE**:

```
$ MDMS DELETE SAVE DENNIS_SAVE
```

MDMS DELETE SCHEDULE

MDMS DELETE SCHEDULE — Deletes a specified schedule definition from the MDMS database.

Format

```
MDMS DELETE SCHEDULE [schedule-name] [, ...]
```

Equivalent **STORAGE** Command: None.

Parameters

schedule-name

Specifies the name of the **SCHEDULE**.

The maximum length of the schedule name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of schedule names may be entered.

Description

The **MDMS DELETE SCHEDULE** command deletes a specified schedule definition from the MDMS database.

Privileges

The request requires **MDMS_DELETE_ALL**.

Restrictions

None.

Qualifiers

/OVERRIDE

The **/OVERRIDE** qualifier allows you to delete the schedule even with outstanding references to it. This is not recommended unless you are sure that the schedule will no longer be needed.

Examples

The following command deletes the schedule definition for `RICHES_SCHEDULE`:

```
$ MDMS DELETE SCHEDULE RICHES_SCHEDULE
```

MDMS DELETE SELECTION

MDMS DELETE SELECTION — Deletes a specified selection definition from the MDMS database.

Format

```
MDMS DELETE SELECTION [selection-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

selection-name

Specifies the name of the selection.

The maximum length of the selection name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of selection names may be entered.

Description

The **MDMS DELETE SELECTION** command deletes a specified selection definition from the MDMS database.

Privileges

The request requires `MDMS_DELETE_ALL`.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/OVERRIDE

The **/OVERRIDE** qualifier allows you to delete the selection even with outstanding references to it. This is not recommended unless you are sure that the selection will no longer be needed

Examples

The following command deletes the `SELECTION` definition for `KASIES_SELECTION`:

```
$ MDMS DELETE SELECTION KASIES_SELECTION
```

MDMS DELETE VOLUME

MDMS DELETE VOLUME — Deletes a specified volume definition from the MDMS volume database.

Format

```
MDMS DELETE VOLUME [volume-id] [, ...]
```

Equivalent `STORAGE` Command: **STORAGE REMOVE VOLUME**

Parameters

volume-id

Specifies the name of the volume. Alternatively, a volume range (separated by a dash) may be specified. A volume range is a numeric range for up to the last five characters of the volume ID. Example ranges are (ABC001-ABC250) and (T20120-T20125).

The maximum length of the volume ID is 6 characters. Valid characters are " ,! , % - ? , A - Z , _ ".

A comma-separated list of volume IDs or ranges may be entered.

Description

The **MDMS DELETE VOLUME** command deletes a specified volume definition from the MDMS volume database. Volumes must be in the `UNINITIALIZED` or `FREE` states prior to being deleted.

Privileges

The request requires `MDMS_DELETE_ALL`, `MDMS_DELETE_VOLUME`, or `MDMS_DELETE_POOL`.

If the user is deleting a volume that is in a pool to which they are authorized, `MDMS_DELETE_POOL` is sufficient. All other volume deletes require `MDMS_DELETE_VOLUME` or `MDMS_DELETE_ALL`.

Restrictions

None.

Qualifiers

None.

Examples

The following command deletes the records for volumes ABC001 to ABC099:

```
$ MDMS DELETE VOLUME ABC001-ABC099
```

MDMS INITIALIZE VOLUME

MDMS INITIALIZE VOLUME — Initializes a volume or a range of volumes by writing the volume ID to tape as the tape label.

Format

```
MDMS INITIALIZE VOLUME [volume-id]
```

Equivalent STORAGE Command: None.

Parameters

volume-id

Specifies the name of the volume to initialize. Alternatively, a volume range (separated by a dash) may be specified. A volume range is a numeric range for up to the last five characters of the volume ID. Example ranges are (ABC001-ABC250) and (T20120-T20125).

There is a maximum of 1000 volumes in a range.

The maximum length of the volume ID is 6 characters. Valid characters are " , ! , " , % - ? , A - Z , _ " .

This parameter is required, except for when the **/JUKEBOX** and **/SLOTS** qualifiers are specified. The **/JUKEBOX** and **/SLOTS** option is valid only for MRD-controlled jukeboxes equipped with a vision system. For all other jukeboxes, the *volume-id* parameter is required.

Description

The **MDMS INITIALIZE VOLUME** command initializes a volume or a range of volumes by writing the volume ID to tape as the tape label. The volume records must be created prior to the initialization. In addition, the volumes must be in the **FREE** or **UNINITIALIZED** states in order to be initialized.

If the volumes are contained in a jukebox, MDMS automatically loads and unloads the volumes, unless the **/DRIVE** qualifier is specified. Otherwise, operator assistance is required.

In order to initialize volumes, the MDMS database server allocates a free drive that can handle the volumes. All volumes in a single command must support the specified media type (or the media type already in the volume record). Do not allocate a drive or load a volume for the initialization—this must be done by MDMS.

Each initialize volume command utilizes a single tape drive for all volumes. To utilize multiple drives, issue multiple initialize volume commands specifying a different volume or slot range, and use the **/NOWAIT** qualifier.

By default, volumes that have a different label than expected and have data written on them are not initialized. However, you can override this check and allow the initialization with the **/OVERRIDE**

qualifier. Note that volumes with labels that are flagged as allocated or in the transition state cannot be initialized under any circumstances.

Privileges

The request requires MDMS_INITIALIZE_ALL or MDMS_INITIALIZE_POOL.

If the user is initializing a volume that is in a pool to which they are authorized, MDMS_INITIALIZE_POOL is sufficient. All other initializations require MDMS_INITIALIZE_ALL.

MDMS_ASSIST is required unless the **/NOASSIST** qualifier is specified.

Restrictions

The **/JUKEBOX** and **/SLOTS** qualifiers must be used together and are incompatible with the *volume-id* parameter.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ASSIST (Default)

/NOASSIST

The default **/ASSIST** qualifier is used to output an operator message if the volume requires operator intervention. You can use **/ASSIST** even if no assistance is needed; MDMS will perform the operation automatically if it can. If you specify **/NOASSIST** and assistance is needed, the operation fails.

This qualifier requires the right MDMS_ASSIST.

/DRIVE=*drive-name*

This qualifier indicates the volume to be initialized is already in the specified drive and should not be loaded or unloaded. This supports an initialize in place. Only one volume may be specified with this option.

/JUKEBOX=*jukebox-name*

This qualifier indicates the volumes to be initialized are in the specified jukebox. A slot range should also be specified with this qualifier. Valid only for MRD jukeboxes equipped with a vision system.

/MEDIA_TYPE=*media-type*

This qualifier modifies the volume record to support only this media type. The volume must already support this media type, but may support others as well. After the initialization, the other media types are removed.

/OVERWRITE

/NOOVERWRITE (Default)

The **/OVERWRITE** qualifier allows initialization if the requested volume label is different from the current label, and the volume has already been initialized and contains data. The default **/NOOVERWRITE** qualifier indicates that the volume should not be initialized if it the label does not

match and the volume contains data. Under no circumstances will the volume be initialized if the volume (according to the current label) is allocated or in the transition state.

/REPLY=*symbol*

The name of the symbol to receive the operator's reply when operator intervention is needed. The symbol will contain the operator reply to the DCL **REPLY/TO** or **REPLY/ABORT** command.

/SLOTS=(*range*[,...])

This qualifier specifies the slots containing volumes to initialize, when used with the **/JUKEBOX** qualifier. Valid only for MRD jukeboxes equipped with a vision system.

/USER=*username*

This qualifier specifies that the volume should be initialized on behalf of this user. The user's UIC is written to the volume label. If the volume is allocated to a user, the allocated user must match the requested user. If **/USER** is not specified, then the username of the calling process is used by default.

This qualifier requires the right MDMS_INITIALIZE_ALL.

/WAIT (Default)**/NOWAIT**

The **/NOWAIT** qualifier returns an informational message indicating that the initialize is being queued. The **/WAIT** qualifier causes the **MDMS INITIALIZE VOLUME** command to wait until the initialization is complete.

Examples

The following command loads, initializes, and unloads volume WOR001 and modifies the volume records to set media type TK85K only:

```
$ MDMS INITIALIZE VOLUME WOR001 /MEDIA_TYPE=TK85K
```

The following command loads, initializes, and unloads the volumes contained in slots 0–10 of jukebox JUKE_1, and allows initialization even if the tapes have unexpected labels and have already been written:

```
$ MDMS INITIALIZE VOLUME/JUKEBOX=JUKE_1 -  
/SLOTS=(0-10)/OVERWRITE
```

The following command loads, initializes, and unloads all 50 volumes in the range ABC001 to ABC050:

```
$ MDMS INITIALIZE VOLUME ABC001-ABC050
```

MDMS INVENTORY JUKEBOX

MDMS INVENTORY JUKEBOX — Verifies the contents of a jukebox. This command is used for both MRD and DCSC controlled jukeboxes.

Format

```
MDMS INVENTORY JUKEBOX [jukebox-name]
```

Equivalent STORAGE Commands: **STORAGE INVENTORY JUKEBOX** and **STORAGE INVENTORY ACS**

Parameters

jukebox-name

Specifies the name of the jukebox.

The maximum length of the jukebox name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

Description

The **MDMS INVENTORY JUKEBOX** command verifies the contents of a jukebox. This command is used for both MRD and DCSC controlled jukeboxes.

For MRD controlled jukeboxes:

- Both the volume range and slot range options are supported, but only one per command.
- If a volume range is specified, all volumes in the range will be inventoried and their slot positions verified and updated as needed. If the volume is not detected in the jukebox, the volume's placement will be according to the **/MISSING** qualifier (or placed in the on-site location by default). The volume range option is only available on jukeboxes supporting a VISION system and with the **/VISION** qualifier specified or defaulted in the command.
- If a slot range is specified, all volumes found in the specified slots are verified and the slot position updated as needed. No other volume records are modified.
- For systems with vision systems, the inventory is performed by querying the jukebox firmware as to slot/drive/port contents. A physical inventory is not performed. Volumes detected in ports will be considered missing (not in the jukebox). It is recommended that all ports are empty during an inventory.
- For small, single-magazine jukeboxes (like the TZ877), each volume is loaded, mounted, examined, and unloaded in order to read the volume label. This can be very time consuming depending on the hardware involved. Inventorying a 7-slot TZ877 jukebox takes about 20 minutes.
- Full or partial inventories are supported.

For DCSC controlled jukeboxes:

- Only the volume range option is supported.
- The inventory simply checks to see if the specified volumes are in the jukebox. This is slightly different behavior than the inventory performed on an MRD-controlled jukebox. If a volume is not in the jukebox, the volume's placement will be according to the **/MISSING** qualifier (or placed in the on-site location by default).

Privileges

The request requires MDMS_INVENTORY_ALL.

Restrictions

The **/SLOTS** and **/VOLUME_RANGE** qualifiers are mutually exclusive.

The **/SLOTS** qualifier is unsupported for DCSC jukeboxes. The **/VOLUME_RANGE** qualifier is required.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/CREATE

This qualifier specifies that any volume that is found that is not defined is created using the **/INHERIT** and/or **/MEDIA_TYPES** qualifiers.

/INHERIT=volume-id

When creating volumes, this qualifier specifies a *volume-id* from which to inherit attributes. The same attributes can be inherited as for the **CREATE VOLUME** command.

/MEDIA_TYPE=media-type

When creating volumes, this qualifier specifies the media type to be used for the volume records.

/MISSING=keyword

This qualifier defines the action to be taken when a volume is missing in the jukebox, when it was defined to be either in a drive or slot in the jukebox. The allowable actions are:

- **DELETE** – Delete the volume record (only valid if volume is in the **FREE** state)
- **IGNORE** – Do nothing
- **MOVE** (Default) – Move the volume to its apparent location (either the on-site location or another jukebox)

/PREINITIALIZED

/NOPREINITIALIZED (Default)

When creating volumes, the volumes are placed in the **FREE** state if **/PREINITIALIZED** is specified. The default is to place them in the **UNINITIALIZED** state.

/SLOTS=(range[,...])

The slots or range of slots to inventory within the jukebox.

/VISION (Default)

/NOVISION

Specifies whether the inventory should be performed with the Vision system on jukeboxes so equipped. A vision inventory reads the internal memory of the jukebox to perform the inventory; a

physical inventory is not performed. If **/NOVISION** is specified, inventory is performed by loading and mounting the volumes. Applicable to MRD jukeboxes equipped with a VISION system only.

/VOLUME_RANGE=(start-id–end-id)

This qualifier specifies the volume range to inventory. A volume range is a numeric range for up to the last five characters of the volume ID. Example ranges are (ABC001-ABC250) and (T20120-T20125).

/WAIT (Default)

/NOWAIT

The **/NOWAIT** qualifier returns an informational message indicating that the inventory is being queued. The **/WAIT** qualifier causes the **MDMS INVENTORY JUKEBOX** command to wait until the inventory is complete.

Examples

The following command inventories the entire jukebox TESTJUKE. Any volumes not in the MDMS volume database will be created with a media type of TK85K. The command prompt is returned after the inventory is queued:

```
$ MDMS INVENTORY JUKEBOX TESTJUKE /CREATE -  
/MEDIA_TYPE=TK85K /NOWAIT
```

The following command inventories the first 10 slots of jukebox TESTJUKE. Any volumes not in the MDMS volume database will be created with a media type of TK85K:

```
$ MDMS INVENTORY JUKEBOX TESTJUKE /CREATE -  
/MEDIA_TYPE=TK85K /SLOTS=(0-9)
```

The following command verifies that volumes SQ0800 through SQ0900 exist in the StorageTek silo. Any volumes found in the silo, but not in the MDMS volume database will be created, with default values. Any volumes found in the MDMS database but not in the silo will be marked in the on-site location of the volume:

```
$ MDMS INVENTORY JUKEBOX SILO_JUKE -  
/VOLUME_RANGE=(SQ0800-SQ0900) /MISSING=MOVE /CREATE
```

MDMS LOAD DRIVE

MDMS LOAD DRIVE — Loads a volume into the specified drive.

Format

```
MDMS LOAD DRIVE [drive-name]
```

Equivalent STORAGE Command: None.

Parameters

drive-name

Specify a drive name, or a logical name.

The maximum length of the drive name or logical name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

Description

The **MDMS LOAD DRIVE** command loads a volume into the specified drive. The command is used primarily for scratch loads when the volume ID is not important—only that a FREE volume is loaded into the drive. The volume may or may not already be created in the MDMS volume database. If it is not created, it may be created using the attributes using the **/INHERIT** or **/MEDIA_TYPE** qualifiers.

Privileges

The request requires MDMS_LOAD_ALL or MDMS_LOAD_SCRATCH.

MDMS_ASSIST is also required unless the **/NOASSIST** qualifier is specified.

The **/CREATE** qualifier requires MDMS_CREATE_POOL if the **/POOL** qualifier is specified and the user is authorized to the pool. All other uses of the **/CREATE** qualifier require MDMS_CREATE_ALL.

Restrictions

The **/NOCHECK** qualifier cannot be used with the **/MOUNT** qualifier.

The **/MOUNT** and **/NOWAIT** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ASSIST (Default)

/NOASSIST

The default **/ASSIST** qualifier is used to output an operator message if the volume requires operator intervention. You can use **/ASSIST** even if no assistance is needed; MDMS will perform the operation automatically if it can. If you specify **/NOASSIST** and assistance is needed, the operation fails.

This qualifier requires the right MDMS_ASSIST

/CREATE

This qualifier specifies that a volume that is not in the MDMS volume database is created in the database. You can also use the **/INHERIT** or **/MEDIA_TYPE** qualifiers to specify the volume's attributes.

This qualifier requires the right MDMS_CREATE_ALL or MDMS_CREATE_POOL if the volume is in a pool owned by the user.

/INHERIT=volume-id

If **/CREATE** is specified, this qualifier allows you to specify an existing volume record to inherit default attributes. The default is that MDMS supplies attribute values you do not on creation.

/MEDIA_TYPE=*media-type*

Specifies the media type assigned to a volume being created with the **/CREATE** qualifier. The media type must be supported by the drive.

/MESSAGE="*message*"

This qualifier allows the application to specify directions to the operator in the OPCOM message associated with the load request. By default, MDMS puts out an appropriate message.

/MOUNT[="*mount-qualifiers*"]

Will cause MDMS to issue a DCL **MOUNT** command once the volume is loaded and the label verified. DCL **MOUNT** qualifiers can also be provided by specifying the qualifiers in a quoted string. By default, the volume is not mounted.

/POOL=*pool-name*

Specifies the volume pool the volume is to be placed in. This qualifier must be specified if the **/CREATE** qualifier is specified and the user only has MDMS_CREATE_POOL privilege, unless a pool is specified in a volume used with the **/INHERIT** qualifier.

/REPLY=*symbol*

The name of the symbol to receive the operator's reply when operator intervention is needed. The symbol will contain the operator reply to the DCL **REPLY/TO** or **REPLY/ABORT** command. The maximum length of a symbol name is 31 characters.

/WAIT (Default)**/NOWAIT**

The **/NOWAIT** qualifier returns an informational message indicating that the load is being queued. The **/WAIT** qualifier causes the **MDMS LOAD DRIVE** command to wait until a volume is loaded.

/WRITE (Default)**/NOWRITE**

The default **/WRITE** qualifier specifies that the volume must be write-enabled when loaded, otherwise a warning is issued after the load. The **/NOWRITE** qualifier specifies that a volume can be successfully loaded with the write protect set on or off.

Examples

The following command loads a volume into drive \$1\$MUA1 and mounts the volume with the **/NOUNLOAD** and **/FOREIGN** qualifiers. The quotes are required:

```
$ MDMS LOAD DRIVE $1$MUA1: /MOUNT="/NOUNLOAD /FOREIGN"
```

MDMS LOAD VOLUME

MDMS LOAD VOLUME — Loads the specified volume into a drive.

Format

```
MDMS LOAD VOLUME [volume-id]
```

Equivalent STORAGE Command: **STORAGE LOAD**

Parameters

volume-id

The maximum length of the volume ID is 6 characters. Valid characters are " ,! , % - ? , A - Z , _ " .

This parameter is required.

Description

The **MDMS LOAD VOLUME** command loads the specified volume into a drive. The volume must already be created in the MDMS database.

If a drive was allocated based on the volume ID, then that drive is chosen for the load. You can also specify a drive on this command, and the volume will be loaded into that drive. The **LOAD VOLUME** command does not allocate the drive to the requesting process.

Privileges

The request requires MDMS_LOAD_ALL, MDMS_LOAD_POOL, or MDMS_LOAD_OWN.

If the user is loading a volume for which they are the owner, MDMS_LOAD_OWN is sufficient. If the user is loading a volume in a pool to which they are authorized, MDMS_LOAD_POOL is required. For all other loads, MDMS_LOAD_ALL is required.

MDMS_ASSIST is also required unless the **/NOASSIST** qualifier is specified.

The **/MOVE** qualifier requires MDMS_MOVE_OWN, MDMS_MOVE_POOL, or MDMS_MOVE_ALL according to the **MOVE** command.

Restrictions

The **/NOCHECK** qualifier cannot be used with the **/MOUNT** qualifier.

The **/MOUNT** and **/WAIT** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ASSIST (Default)

/NOASSIST

The default **/ASSIST** qualifier is used to output an operator message if the volume requires operator intervention. You can use **/ASSIST** even if no assistance is needed; MDMS will perform the operation automatically if it can. If you specify **/NOASSIST** and assistance is needed, the operation fails.

This qualifier requires the right MDMS_ASSIST.

/CHECK (Default)
/NOCHECK

Compares the physical ANSI label on the tape against the label for the volume ID. If the physical ANSI label does not match the label for the volume ID, operator intervention is required to resolve the conflict. The default is **/CHECK**.

/DRIVE=drive-name

This qualifier specifies the name of the drive in which to load the volume.
This qualifier is required unless a drive has been allocated for this volume (i.e. **ALLOCATE DRIVE/VOLUME=volume-id**).

/MESSAGE="message"

This qualifier allows the application to specify directions to the operator in the OPCOM message associated with the load request.

/MOUNT[="mount-qualifiers"]

Will cause MDMS to issue a DCL **MOUNT** command once the volume is loaded and the label verified. DCL **MOUNT** qualifiers can also be provided by specifying the qualifiers in a quoted string. By default, the volume is not mounted.

/MOVE (Default)
/NOMOVE

The default **/MOVE** qualifier, used with **/ASSIST**, allows the load request to generate a move request to move a volume from a remote location to the drive or associated jukebox. The move will generate an OPCOM move request which that must be satisfied before the load request can continue. If **/NOMOVE** is specified and a move is required, the load request fails with an error.

/REPLY=symbol

The name of the symbol to receive the operator's reply when operator intervention is needed. The symbol will contain the operator reply to the DCL **REPLY/TO** or **REPLY/ABORT** command. The maximum length of a symbol name is 31 characters.

This qualifier is only applicable when the **/ASSIST** qualifier is specified.

/WAIT (Default)
/NOWAIT

The **/NOWAIT** qualifier returns an informational message indicating that the load is being queued. The **/WAIT** qualifier causes the **MDMS LOAD VOLUME** command to wait until the volume is loaded.

/WRITE (Default)
/NOWRITE

The default **/WRITE** qualifier specifies that the volume must be write-enabled when loaded, otherwise a warning is issued after the load. The **/NOWRITE** qualifier specifies that a volume can be successfully loaded with the write protect set on or off.

Examples

The following command loads volume ABC010 into a previously-allocated drive, mounts the volume with the **/NOUNLOAD** qualifier, and verifies the volume label. The quotes are required:


```
$ MDMS LOAD VOLUME ABC010 /MOUNT="/NOUNLOAD"
```

The following command loads volume ABC020 into drive \$1\$MUA1 : and does not check the on-tape volume label:

```
$ MDMS LOAD VOLUME ABC020 /NOCHECK /DRIVE=$1$MUA1:
```

The following command loads volume ABC020 into an allocated drive, checks the on-tape volume label, but does not wait for the load to complete before returning the command prompt:

```
$ MDMS LOAD VOLUME ABC020 /NOWAIT
```

MDMS MOVE MAGAZINE

MDMS MOVE MAGAZINE — Moves a magazine from one location to another. The magazine must be created prior to the move.

Format

```
MDMS MOVE MAGAZINE [magazine-name] [destination]
```

Equivalent STORAGE Commands: **STORAGE EXPORT MAGAZINE** and **STORAGE IMPORT MAGAZINE**

Parameters

magazine-name

Defines the name of the magazine to move. Only one magazine may be moved per command. If the **/SCHEDULE** qualifier is specified, the *magazine-name* may be the wildcard *, which means all scheduled magazines.

The maximum length of the magazine name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

destination

Specifies the name of the destination object. The destination object can be a location or a jukebox. The destination is optional when the **/SCHEDULE** qualifier is entered, but required for commands when **/SCHEDULE** is not entered. The default destination is the appropriate off-site or on-site location specified in the magazine object.

Description

The **MDMS MOVE MAGAZINE** command moves a magazine from one location to another. Magazines can be moved between outside locations, and from an outside location to and from a jukebox.

When the **/ASSIST** qualifier is specified, a series of OPCOM messages may be displayed asking the operator to move magazines between locations, or in and out of the jukebox. No slots or spaces are displayed in the OPCOM message, but the operator should issue a **SHOW MAGAZINE/FULL** command for the associated magazines to determine slots, spaces, or positions involved in the move. These are shown in the placement field.

Privileges

The request requires MDMS_MOVE_ALL.

MDMS_ASSIST is also required unless the **/NOASSIST** qualifier is specified.

Restrictions

The **/POSITION** and **/START_SLOT** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ASSIST (Default)

/NOASSIST

The default **/ASSIST** qualifier is used when the magazine has not been physically moved, and an operator needs to physically move the magazine. Use **/NOASSIST** if the magazine has already been physically moved, or if you plan to move it yourself.

This qualifier requires the right MDMS_ASSIST.

/OFFSITE

When moving to a location that is not already specified in the magazine record, this qualifier specifies that the location is an off-site location, and the magazine placement is off-site, following the move. By default, an unspecified location is on-site.

/POSITION=(*position*)

The **/POSITION** qualifier specifies that the magazine is being moved into a jukebox, and specifying the jukebox position that the magazine is being moved to. The **/POSITION** qualifier or **/START_SLOT** qualifier must be specified when moving a magazine into a multi-magazine jukebox.

In addition, the **/POSITION** qualifier can be used to transfer a magazine between positions inside a jukebox (if physically possible). The position parameter is in the format (tower, face, level).

The 'tower, face, and level' specification represents the relative number of the tower, face, and level, starting from 0.

For example, for the absolute jukebox slot of zero, the corresponding position is (0,0,0). The next position in the jukebox would be (0,0,1) and so on, according to the topology defined for the jukebox. Valid for MRD-controlled jukeboxes only.

/REPLY=*symbol*

The name of the symbol to receive the operator's reply when operator intervention is needed. The symbol will contain the operator reply to the DCL **REPLY/TO** or **REPLY/ABORT** command. The maximum length of a symbol name is 31 characters.

This qualifier is only applicable when the **/ASSIST** qualifier is specified.

/SCHEDULE[=*keyword*]

The **/SCHEDULE** qualifier selects magazines whose off-site or on-site date has "expired" and the magazine is not in the new location. If both dates have expired, the later of the two dates is used. The optional keywords on the **/SCHEDULE** qualifier may be OFFSITE or ONSITE.

If the OFFSITE keyword is used, then only those magazines scheduled to be moved off site are selected. If the ONSITE keyword is used, then only those magazines scheduled to be moved on site are selected. If the keyword is omitted, then the magazines scheduled to be moved both on site and off site are selected.

/START_SLOT=*number*

This qualifier specifies the starting jukebox slot when the magazine is placed in a jukebox. The default is zero. Valid for MRD-controlled jukeboxes only.

/SPACES=(*range*[,...])

The **/SPACES** qualifier specifies the spaces in a non-jukebox location to move the magazine to. You should specify a single space if you are moving a single magazine. If you issue **/SCHEDULE**, each magazine will be moved to the next space in the range in order.

/WAIT (Default)

/NOWAIT

The **/NOWAIT** qualifier returns an informational message indicating that the move is being queued. The **/WAIT** qualifier causes the **MDMS MOVE MAGAZINE** command to wait until the magazine is moved.

Examples

The following command moves magazine MYMAG01 from its current location into jukebox JUKE_1 at start slot of zero:

```
$ MDMS MOVE MAGAZINE MYMAG01 JUKE_1
```

The following command moves magazine MYMAG01 from its current location to jukebox JUKE_2 in position (2, 0, 1). This is equivalent to tower 2, face 0, level 1:

```
$ MDMS MOVE MAGAZINE MYMAG01 JUKE_2 /POSITION=(2, 0, 1)
```

The following command moves magazine MYMAG02 from its current position out of the jukebox to location ROOM_100 in space 23:

```
$ MDMS MOVE MAGAZINE MYMAG02 ROOM_100 /SPACES=23
```

The following command moves magazine MAG002 from its current location into jukebox JUKE_1 in start slot of 11:

```
$ MDMS MOVE MAGAZINE MAG002 JUKE_1 /START_SLOT=11
```

The following command moves all magazines whose off-site date has expired to the magazine's off-site location:

```
$ MDMS MOVE MAGAZINE * /SCHEDULE=OFFSITE
```

The following command moves all magazines whose off-site and on-site dates have expired to the magazine's off-site and on-site locations respectively:

```
$ MDMS MOVE MAGAZINE * /SCHEDULE
```

MDMS MOVE VOLUME

MDMS MOVE VOLUME — Moves a volume (or volumes) from one location to another. The volumes must be created prior to the move.

Format

```
MDMS MOVE VOLUME [volume-id] [destination]
```

Equivalent **STORAGE** Commands: **STORAGE BIND**, **STORAGE EXPORT ACS**, **STORAGE EXPORT VOLUME**, **STORAGE IMPORT ACS**, **STORAGE IMPORT VOLUME**, and **STORAGE UNBIND**

Parameters

volume-id

Defines the name of the volume to move. A *volume-id* may be a single volume, a list of volumes, a volume range (separated by a dash), or a list of volume ranges. A volume range is a numeric range for up to the last five characters of the volume ID. Example ranges are (ABC001- ABC250) and (T20120-T20125).

A *volume-id* may also be a wildcard *, which means all scheduled volumes. The wildcard is only valid when the **/SCHEDULE** qualifier is specified. This parameter is required.

The maximum length of the volume ID is 6 characters. Valid characters are " ,! ,", %-?, A-Z, _".

destination

Specifies the name of the destination object. The destination object can be a location, magazine, or a jukebox. The destination is optional when **/SCHEDULE** is entered, but required for commands when **/SCHEDULE** is not entered. For **/SCHEDULE**, the default destination is the appropriate off-site or on-site location specified in the volume object.

Description

The **MDMS MOVE VOLUME** command moves a volume or volumes from one location to another. The volumes must be created prior to the move. Multiple volumes can be moved, where appropriate, in a single command.

When the **/ASSIST** qualifier is specified, a series of OPCOM messages may be displayed asking the operator to move volumes between locations, or in and out of a jukebox or magazine. No slots or spaces are displayed in the OPCOM message, but the operator should issue a **SHOW VOLUME/FULL** command for the associated volumes to determine slots or spaces involved in the move. These are shown in the placement field.

When moving volumes into a jukebox, the specification of slots is optional. If not supplied, MDMS will apply default values and move the volumes into empty slots.

This command is not used to load volumes into or out of drives.

Privileges

The request requires MDMS_MOVE_ALL, MDMS_MOVE_POOL, or MDMS_MOVE_OWN.

If the user is moving a volume for which they are the owner, MDMS_MOVE_OWN is sufficient. If the user is moving a volume in a pool to which they are authorized, MDMS_MOVE_POOL is required. All other moves require MDMS_MOVE_ALL.

MDMS_ASSIST is also required unless the **/NOASSIST** qualifier is specified.

The **/DESCRIPTION** qualifier requires MDMS_SET_ALL, MDMS_SET_POOL, or MDMS_SET_OWN.

Restrictions

The **/[NO]ASSIST** qualifier cannot be used with the **/PORT** qualifier.

Qualifiers

/ASSIST (Default)

/NOASSIST

The default **/ASSIST** qualifier is used to output an operator message if the volume requires operator intervention. You can use **/ASSIST** even if no assistance is needed; MDMS will perform the operation automatically if it can. If you specify **/NOASSIST** and assistance is needed, the operation fails.

This qualifier requires the right MDMS_ASSIST.

/CAP=*cap-id*

If moving a volume to/from a silo, this qualifier specifies the Cartridge Access Port identifier into which the volume is being physically injected or ejected. Required when moving a volume into or out of a silo. Valid for DCSC-controlled jukeboxes only.

/DESCRIPTION="*text*"

Modifies comments about the object in the volume record. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the existing description, specify "".

This qualifier requires the right MDMS_SET_OWN, MDMS_SET_POOL, or MDMS_SET_ALL, depending on the user's ownership of the volume.

/NOPHYSICAL

This qualifier requests that no internal jukebox calls (through DCSC or MRD) are made to implement the move. When used with the **/NOASSIST** qualifier, the effect of the **MOVE VOLUME**

command is to simply update the database to the new location. When used with the default of **/ASSIST**, a single OPCOM message is displayed to move the volumes, but when the message expires, the database is automatically updated.

/OFFSITE

When moving to a location that is not already specified in the volume record, this qualifier specifies that the location is an off-site location, and the volume placement is off site following the move. By default, an unspecified location is on site.

/PORT

When addressing the **MOVE VOLUME** request using the **/PORT** qualifier, MDMS independently moves volumes to the outports and considers the request as complete (based on the availability of free ports). In addition, MDMS also displays the volume move status.

When the **/PORT** qualifier is used, in order to complete the **MOVE VOLUME** request, MDMS does not indefinitely wait for the operator to remove the volumes from the outports. Instead, irrespective of the success or failure of the command, MDMS displays an OPCOM and ends the request. All the OPCOMS are displayed once and do not require user acknowledgement. The following are the different statuses for the **MOVE VOLUME** request:

- If the **MOVE VOLUME** request is successful, MDMS updates the volume placement and displays an OPCOM indicating the success status.
- If the **MOVE VOLUME** request is failing because the outport is full, then MDMS does the following:
 1. Displays an OPCOM listing the volumes that were not moved.
 2. Displays the error message "%MDMS-E-NOVOLSMOVED, no volumes were moved", indicating that none of the volumes were moved.
 3. Ends the **MOVE VOLUME** request.
- If the **MOVE VOLUME** request is incomplete because the operator tried to move more volumes than the number of available ports, then MDMS does the following:
 1. Moves volumes only for the number of free ports available.
 2. Displays an OPCOM listing the volumes that were moved and those that could not be moved.
 3. Displays the error message "%MDMS-E-MOVEINCOMPL, move is incomplete", indicating that the movement of volumes was incomplete.
 4. Ends the **MOVE VOLUME** request.

/REPLY=*symbol*

The name of the symbol to receive the operator's reply when operator intervention is needed. The symbol will contain the operator reply to the DCL **REPLY/TO** or **REPLY/ABORT** command. The maximum length of a symbol name is 31 characters.

This qualifier is only applicable when the **/ASSIST** qualifier is specified.

/SCHEDULE[=*keyword*]

The **/SCHEDULE** qualifier selects volumes whose off-site or on-site date has "expired" and the volumes are not in the new location. If both dates have expired, the later of the two dates is used. The optional keywords on the **/SCHEDULE** qualifier may be OFFSITE or ONSITE.

If the OFFSITE keyword is used, then only those volumes scheduled to be moved off site are selected. If the ONSITE keyword is used, then only those volumes scheduled to be moved on site are selected. If the keyword is omitted, then volumes scheduled to be moved both on site and off site are selected.

/SLOTS=(*range*[,...])

The **/SLOTS** qualifier specifies that the volume is being moved into a jukebox or magazine and specifying the slot range, or slot list, that the volumes are being moved to. If not specified, MDMS selects free slot locations for the volumes. In addition, the **/SLOTS** qualifier can be used to transfer volumes between slot locations inside a jukebox or magazine. If moving a single volume, specify a single slot. If moving multiple volumes, the slots in the range are assigned in order. If any of the specified slots are full, MDMS selects alternate slots. Valid for MRD-controlled jukeboxes only.

/SPACES=(*range*[,...])

The **/SPACES** qualifier specifies the spaces in a non-jukebox location to move the volumes to.

/WAIT (Default)**/NOWAIT**

The **/NOWAIT** qualifier returns an informational message indicating that the move is being queued. The **/WAIT** qualifier causes the **MDMS MOVE VOLUME** command to wait until the command is complete.

Examples

The following command moves the volume VOL1 to one of the ports that is empty. The move request is successful:

```
$ MDMS MOVE VOLUME VOL1 TEST_LOC/PORT
%MDMS-I-TEXT, Volumes : BTY096 moved to port.
```

The following command tries to move the volume VOL2 when all the ports are full. The move request is unsuccessful:

```
$ MDMS MOVE VOLUME VOL2 TEST_LOC/PORT
%MDMS-I-TEXT, Volumes : Test_Vol2 not moved to port.
%MDMS-E-NOVOLSMOVED, no volumes were moved
```

The following command tries to move the volumes VOL1 and VOL2 when only one port is free. The move request is incomplete as only one port is free:

```
$ MDMS MOVE VOLUME VOL1-VOL2 TEST_LOC/PORT
%MDMS-I-TEXT, Volumes : Vol1 moved to port. Volumes:: Vol2 not moved to port.
%MDMS-E-MOVEINCOMPL, move is incomplete
```

The following command moves volume ABC001 from its current location to jukebox JUKE_2 at slot 20:

```
$ MDMS MOVE VOLUME ABC001 JUKE_2 /SLOTS=20
```

The following command moves the volume AGW081 to location SHELF at space 42:

```
$ MDMS MOVE VOLUME AGW081 SHELF /SPACES=42
```

The following command moves volumes FLI050 to FLI056 into magazine MAG001 at slots (0-6). Note that this replaces the old **STORAGE BIND** command:

```
$ MDMS MOVE VOLUME FLI050-FLI056 MAG001 /SLOTS=(0-6)
```

The following command moves all volumes whose off-site date has expired to the volume's off-site location:

```
$ MDMS MOVE VOLUME * /SCHEDULE=OFFSITE
```

The following command moves all volumes in the range ABC001-ABC100 whose off-site date has expired to the volume's location ARGUS_VAULT:

```
$ MDMS MOVE VOLUME ABC001-ABC100 ARGUS_VAULT /SCHEDULE=OFFSITE
```

The following command moves all volumes whose off-site and on-site dates have expired to the default off-site and on-site locations respectively:

```
$ MDMS MOVE VOLUME * /SCHEDULE
```

The following command move the volumes ALS100-ALS150 to the DCSC jukebox WOLFCREEK using CAP1:

```
$ MDMS MOVE VOLUME ALS100-ALS150 WOLFCREEK /CAP=1
```

MDMS REPORT VOLUME

MDMS REPORT VOLUME — Reports on volume objects.

Format

```
MDMS REPORT VOLUME [field[=value]] [, ...]
```

Equivalent STORAGE Command: **REPORT VOLUME**

Parameters

field

The *field* parameter selects attributes to display for the object.

For any string value, you can use a wildcard character. The * character provides a wild card for any number/length of characters. The % character is a wild card character for one character.

For date and numeric values, a range may be specified by entering a two value list in parentheses. For example:

```
SCRATCH_DATE=(01-Jun-2002, 01-Jul-2002)
MOUNT_COUNT=(100,9999)
```

The first value is interpreted as a low value, and the second value is interpreted as a high value. All values in the range are selected for display.

At least one field must be defined for a report to be generated, except when the **/FORECAST** or **/SUMMARY** qualifier is entered.

Field	Meaning	Value Format	Example Output
ACCOUNT	Owner's account	String	ENGINEERING
ALLOCATION_DATE	Last allocation	Date	18-MAY-2001
AVAILABLE_STATE	State the volume will be put in when it becomes available	Keyword	ALLOCATED
BLOCK_FACTOR	Number of records in one block	Numeric	20
BRAND	Brand name of a volume	String	VSI
CLEANED_DATE	Last date the volume was cleaned	Date DD-MMM-YYY	18-MAY-2001
CREATION_DATE	Date volume was entered in the database	Date DD-MMM-YYY	18-MAY-2001
DEALLOCATION_DATE	Date volume was last deallocated	Date DD-MMM-YYY	18-MAY-2001
DESCRIPTION	Description	String	DRA2 BACKUP
DRIVE	Name of tape drive	String	\$1\$MUA560
ERROR_COUNT	Number of I/O errors since a volume was last cleaned	Numeric	20
FORMAT	Recording format	Keyword	BACKUP
FREED_DATE	Date volume was last freed	Date DD-MMM-YYY	18-MAY-2001
INITIALIZED_DATE	Date volume was initialized	Date DD-MMM-YYY	18-MAY-2001
JOB	Name of the job that wrote the volume	String	ABS
JUKEBOX	Name of the jukebox	String	TL812_JUKE
LAST_ACCESS_DATE	Date volume was last accessed	Date DD-MMM-YYY	18-MAY-2001
MAGAZINE	Name of magazine	String	ENG_MAG
MEDIA_TYPE	Type of media	String	TK89
MOUNT_COUNT	Number of times a volume has been loaded by MDMS	Numeric	20
NEXT_VOLUME	Next volume in a set	String	AGW200
OFFSITE_DATE	Date a volume is to be taken off site	Date DD-MMM-YYY	18-MAY-2001

Field	Meaning	Value Format	Example Output
OFFSITE_LOCATION	Location where volume resides when it is off site	String	VAULT
ONSITE_DATE	Date a volume is to be returned on site	Date DD-MMM-YYY	18-MAY-2001
ONSITE_LOCATION	Location where the volume resides when it is on site	String	ROOM_256
OWNER_UIC	Owner's user identification code	Standard Open- VMS UIC format	[311,311][ABS]
PLACEMENT	Current placement of the volume	Keyword	JUKEBOX
PLACENAME	Current placement name of the volume	String	NODE01_JUKE
POOL	Volume's pool name	String	ENGINEERING
PREVIOUS_VOLUME	Previous volume in a volume set	String	AGW201
PURCHASED_DATE	Volume's purchase date	Date DD-MMM-YYY	18-MAY-2001
PROTECTION	Access protection code	Standard Open- VMS protection codes	"S:RWED, O:RWED, G:E, W:E"
SCRATCH_DATE	Date volume is to be freed	Date DD-MMM-YYY	18-MAY-2001
SLOT	Number of the jukebox slot in which the volume resides	Numeric	20
SPACE	A non-jukebox space in the specified location where the volume resides	String	A120
STATE	The state of the volume	Keyword	ALLOCATED
TIMES_CLEAVED	Number of times the volume has been cleaned	Numeric	10
TRANSITION_TIME	The time a volume stays in the transition state before going to the free state	Time DDDDHH:MM:SS	15-00:00:00
USER	Owner's user name	String	SMITH
VOLUME	Volume ID	String	AGW200

Description

The **MDMS REPORT VOLUME** command generates a report on the selected volumes. The contents of the report contain fields as specified in the field parameters. Each selected field is displayed unless the **/NOPRINT** field qualifier is specified. The report can be sorted on the field by using the **/SORT** qualifier on a single field (only one sort key supported per command). In addition, the width of each field can be specified with the **/WIDTH** qualifier. The default width is specific for each field size.

Privileges

The request requires **MDMS_SHOW_ALL**, **MDMS_SHOW_POOL**, or **MDMS_SHOW_OWN**. If the user only has **MDMS_SHOW_OWN**, only those allocated volumes owned by the user will be included in the report. If the user has **MDMS_SHOW_POOL**, only those volumes in pools for which the user is authorized will be displayed. If the user has **MDMS_SHOW_ALL**, all potential volumes matching the selection criteria will be displayed.

Restrictions

The **/USER_NAME** qualifier can only be used with the **/FORECAST** and **/SUMMARY** qualifiers. For all other types of report use the selection value of the **USER** field.

Fields cannot be used with the **/FORECAST** and **/SUMMARY** qualifiers.

The **/FORECAST** and **/SUMMARY** qualifiers are mutually exclusive.

Fields must be specified with the **/SCHEDULE** qualifier.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/FORECAST

Displays all volumes allocated to a user sorted by their scratch date. If the **/USER_NAME** qualifier is not specified, only those volumes allocated to you are displayed. Do not specify any fields.

/HEADER (Default)

/NOHEADER

The **/HEADER** qualifier specifies that the report should contain header information. This is the default. If you do not wish to have header information, then specify **/NOHEADER**.

/OUTPUT=*file-spec*

Directs the report output to the specified file instead of **SY\$OUTPUT**, which is usually the command screen.

/NOPRINT

The **/NOPRINT** field qualifier specifies that a field defined as a selection criteria is not included in the report.

SCHEDULE[=*keyword*]

The **/SCHEDULE** qualifier selects volumes whose off-site or on-site date has "expired" and the volumes are not in the new location. If both dates have expired, the later of the two dates is used. The optional keywords on the **/SCHEDULE** qualifier may be OFFSITE or ONSITE.

If the OFFSITE keyword is used, then only those volumes scheduled to be moved off site are selected. If the ONSITE keyword is used, then only those volumes scheduled to be moved on site are selected. If the keyword is omitted, then volumes scheduled to be moved both on site and off site are selected.

/SORT

This is a field qualifier, and is used as a sort key. Only one field can be used as a sort key, and no value may be specified.

/SUMMARY

Displays all volumes allocated to a user and sorted by volume ID. If the **/USER_NAME** qualifier is not specified, only those volumes allocated to you are displayed. Do not specify any fields.

/TITLE=*text*

Specifies the title of the report. If the text contains spaces, it must be enclosed in quotation marks. The maximum length of the title is 80 characters.

/USER_NAME=*username*

Selects volumes owned by this user. This qualifier can only be used with the **/FORECAST** and **/SUMMARY** qualifiers. If **/USER_NAME** is not specified, only those volumes for the present user are selected.

/WIDTH=*number*

A field qualifier, this specifies the number of characters to display for a field. If not specified, a default width is applied to each field type, and excess characters may be truncated.

Examples

The following command prints a report for all volumes in pool ABS_POOL and prints out the volume ID, pool name, allocation state, and scratch date:

```
$ MDMS REPORT VOLUME VOLUME, POOL=ABS_POOL, STATE, -  
SCRATCH_DATE
```

The following command prints a report for all allocated volumes and prints out the volume ID and scratch date:

```
$ MDMS REPORT VOLUME VOLUME, STATE=ALLOCATED/NOPRINT, -  
SCRATCH_DATE
```

The following command prints a report for all allocated volumes and prints out the volume ID, allocation state, and owner (maximum of 10 characters), sorted by owner:

```
$ MDMS REPORT VOLUME VOLUME, STATE=ALLOCATED, -  
OWNER/SORT/WIDTH=10
```

The following command prints a report for all volumes allocated to user SMITH and prints out the volume ID, allocation date, scratch date, format, and description fields, sorted by scratch date:

```
$ MDMS REPORT VOLUME /FORECAST /USER_NAME=SMITH
```

MDMS SET ARCHIVE

MDMS SET ARCHIVE — Modifies an existing archive definition in the MDMS database. Archives were formerly known as storage classes, archive classes and storage policies. Archive is the new common name.

Format

```
MDMS SET ARCHIVE [archive-name] [, ...]
```

Equivalent ABS Command: **SET STORAGE_CLASS**

Parameters

archive-name

Specifies the name of the archive.

The maximum length of the archive name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of archive names may be entered.

Description

The **MDMS SET ARCHIVE** command modifies an existing archive definition in the MDMS database. An archive describes where data is saved and its catalog information.

Privileges

The request requires MDMS_SET_ALL.

The **/VOLUME_SETS** qualifier also requires MDMS_SET_PROTECTED since this attribute is normally managed by MDMS. You should not modify this attribute unless you are trying to recover from an abnormal situation.

Restrictions

This command is restricted to nodes with an ABS license.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The **/EXPIRATION_DATE** and **/RETENTION_DAYS** qualifiers are mutually exclusive.

The following combinations of qualifiers are not allowed based on the archive type:

DISK:	/CONSOLIDATION	TAPE:	/DESTINATION
	/DRIVES		

/LOCATION
/MEDIA_TYPE
/POOL
/VOLUME_SETS

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/ARCHIVE_TYPE=*keyword*

The archive type specifies the type of archive media associated with the archive. The options are:

- DISK – Files are saved to or restored from a disk. The destination may be filled in to specify the disk and directory where the archive data resides.
- TAPE – Files are saved to or restored from a tape volume and MDMS is used to manage the tape volume.

The default is TAPE.

/CATALOG=(NAME=*catalog-name*, [NO]NODES=(*node*[,...]))

The **/CATALOG** qualifier specifies the name of the catalog used by the archive, and a list of nodes and/or groups that can directly access the catalog. Specify a local catalog name representing a catalog object in the MDMS database. With the NODES keyword, you may specify nodes, groups or a combination thereof.

If you specify NONODES, the local execution node on which a save or restore is performed is used as the catalog node.

By default, the catalog name "ABS_CATALOG" is used. For an archive named "DISASTER_RECOVERY" you may specify an empty name using "".

The default for nodes is NONODES.

/CONSOLIDATION=(*option*[,...])

The **/CONSOLIDATION** qualifier specifies the criteria under which new volume sets are created for this archive. You may specify one or more of the following options:

- **INTERVAL=*delta-time*** – The time to elapse from the creation of the last volume set to the creation of a new volume set. To specify an unlimited consolidation interval, specify NOINTERVAL.
- **SAVESETS=*number*** – The maximum number of savesets to exist on a volume set before a new volume set is created. Zero means no limit.
- **VOLUMES=*number*** – The maximum number of volumes which may be added to the current volume set. Zero means no limit.

A new volume set is created when any one of the consolidation criteria expire. By default, the consolidation criteria is set to the following:

```
INTERVAL=7-00:00:00, SAVESETS=0, VOLUMES=0
```

If no consolidation criteria are specified, new volume sets are not created unless an error occurs. This qualifier is allowed only with archive type TAPE.

/DESCRIPTION="*text*"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/DESTINATION=*disk*:[*dir*]

If the archive type is DISK, the **/DESTINATION** qualifier specifies the disk and directory where the archive data resides. If no destination is specified for archive type DISK, the following is used by default:

```
ABS$ROOT:[000000]
```

This qualifier is allowed only with archive type DISK.

/DRIVES=(*drive-name*[,...])
/NODRIVES (Default)

The **/DRIVES** qualifier specifies a list of tape drives to be used by the archive. Specify one or more drive names. By default, no drives are specified, meaning that any drive satisfying other compatibility criteria may be used.

This qualifier is allowed only with archive type TAPE.

/EXPIRATION_DATE=*absolute-time*
/NOEXPIRATION_DATE

The **/EXPIRATION_DATE** qualifier specifies the date that saved data will expire. Specify an absolute or delta date/time, or a reserved word such as TODAY or TOMORROW. Alternatively, use the relative **/RETENTION_DAYS** qualifier to specify the number of days the saved data should be retained. **/NOEXPIRATION_DATE** means that the data is retained indefinitely.

By default, data is retained for 365 days.

/LOCATION=*location-name*
/NOLOCATION

The **/LOCATION** qualifier specifies the location to be used to allocate a drive to load a scratch volume (for example, **HELP MDMS ALLOCATE DRIVE/LOCATION**). Specify a location name specified in the database. This qualifier is applicable only for archive type TAPE. **/NOLOCATION** removes location as a selection criterion for volume allocation.

By default, the domain default on-site location is used.

/MAXIMUM_SAVES=*number*

The **/MAXIMUM_SAVES** qualifier specifies the maximum number of save operations that can execute simultaneously using this archive. For example, you can create three save requests and schedule them to run simultaneously, assuming that there are enough resources (e.g. drives) available. Valid values are 1–36. The default is 1.

/MEDIA_TYPE=*media-type*

The **/MEDIA_TYPE** qualifier specifies the media type of tape volumes used for this archive. By default, the domain default media type is used if the archive type is TAPE, otherwise none.

This qualifier is allowed only with archive type TAPE.

/OWNER_NAME=*node::username*
/OWNER_NAME=*group::username*
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an

owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *cluster-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/POOL=pool-name

/NOPOOL (Default)

The **/POOL** qualifier specifies the pool from which volumes will be allocated. Specify a pool name that exists in the database. If you specify **/NOPOOL**, then only volumes with no pool definition may be used (equivalent to the SCRATCH pool).

By default, no pool is defined.

This qualifier is allowed only with archive type TAPE.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/RETENTION_DAYS=number

The **/RETENTION_DAYS** qualifier specifies the number of days to retain data for this archive. Either this qualifier or **/EXPIRATION_DATE** may be used to specify this information. If neither an expiration date nor retention days value is specified, then the data is retained indefinitely.

By default, a retention of 365 days is used.

/VOLUME_SETS=(volume-id[,volume-id])

/NOVOLUME_SETS

The **/VOLUME_SETS** qualifier specifies the active volume sets being used (for write) by the archive. Each *volume-id* should represent the first volume in the volume set.

/NOVOLUME_SETS clears the volume sets, and forces a new volume set to be allocated when the archive is next used.

Setting of volume sets is not recommended as ABS normally assigns volume sets to archives. This is a protected field.

This qualifier requires the right MDMS_SET_PROTECTED.

This qualifier is allowed only with archive type TAPE.

Examples

The following command modifies an archive USER_BACKUPS, removing access control for user SMITH on all nodes and user JONES on node NOD001. It also modifies the consolidation criterion to 100 savesets per volume set and specifies an expiration date of 01-JUL-2001:

```
$ MDMS SET ARCHIVE USER_BACKUPS -  
_ $ /ACCESS_CONTROL=(USER=*::SMITH/REMOVE -  
_ $ /CONSOLIDATION=SAVESETS=100 -  
_ $ EXPIRATION_DATE=01-JUL-2001
```

MDMS SET CATALOG

MDMS SET CATALOG — Modifies an existing catalog definition in the local ABS database.

Format

```
MDMS SET CATALOG [catalog-name] [, ...]
```

Equivalent ABS Command: **RUN SYS\$SYSTEM:ABS\$CATALOG_OBJECT**

Parameters

catalog-name

Specifies the name of the catalog.

The maximum length of the catalog name is 26 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of catalog names may be entered.

Description

The **MDMS SET CATALOG** command modifies an existing catalog definition in the MDMS database. A catalog describes how, where and what information about saved data is recorded.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- **SET** – Allows setting the attributes of the object

- **SHOW** – Allows showing the object
- **DELETE** – Allows deleting the object
- **READ** – Allows restoring data using the object
- **WRITE** – Allows saving data using the object
- **EXECUTE** – Allows execution operations using the object
- **CONTROL** – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The user-id should be in the format `node::username` or `group::username`. Wildcards are supported in the `node`, `group` and `username` portions. Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/NODE=*node-name*

The **/NODE** qualifier specifies on which node the catalog will be modified. The default is the node on which the **SET CATALOG** command is entered.

/OWNER_NAME

/OWNER_NAME=*node::username*

/OWNER_NAME=*group::username*

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/STAGING (Default)

/NOSTAGING

The **/STAGING** qualifier turns on the use of a staging catalog during saves. Using a staging catalog can greatly speed up a save operation, and deferring update of the full catalog to an off-line operation.

However, a restore cannot be performed until the full catalog is updated. Use of staging is highly recommended. The **/NOSTAGING** qualifier turns off use of a staging catalog. Staging is only used on catalog types FILES and VOLUME_SETS.

Examples

The following command disables staging on catalog JOE_CATALOG on node MYJOE:

```
$ MDMS SET CATALOG JOE_CATALOG /NOSTAGING /NODE=MYJOE
```

MDMS SET DOMAIN

MDMS SET DOMAIN — Modifies the MDMS Domain object. The MDMS domain contains attributes affecting all nodes, devices and locations that comprise an MDMS domain.

Format

```
MDMS SET DOMAIN
```

Equivalent STORAGE Command: None.

Parameters

None.

Description

The **MDMS SET DOMAIN** command modifies the MDMS domain. The MDMS domain contains attributes affecting all nodes, devices, and locations that comprise an MDMS domain.

Privileges

The request requires MDMS_SET_ALL. The qualifiers associated with setting of privilege rights also require MDMS_SET_RIGHTS.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ABS_RIGHTS

/NOABS_RIGHTS (Default)

The **/ABS_RIGHTS** qualifier enables a certain set of MDMS rights when users have an ABS right set in the UAF record. The default **/NOABS_RIGHTS** qualifier does not give a users with ABS rights any additional MDMS rights.

This qualifier requires the right MDMS_SET_RIGHTS.

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/APPLICATION_RIGHTS[=(*right*[,...])

/NOAPPLICATION_RIGHTS

The **/APPLICATION_RIGHTS** qualifier sets the low-level rights associated with the high-level right MDMS_APPLICATION. If the qualifier is specified with no value, a default set of rights is set.

The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace rights in the list, rather than adding them by default. Use **/NOAPPLICATION_RIGHTS** to remove all rights.

This qualifier requires the right MDMS_SET_RIGHTS.

/CHECK_ACCESS

/NOCHECK_ACCESS (Default)

The **/CHECK_ACCESS** qualifier enables access control validation for all MDMS requests. When enabled, MDMS validates access to objects according to the operation being requested. If there are no access control entries on the object, access is either granted or rejected depending on the default access state.

The default **/NOCHECK_ACCESS** does not perform any access control validation in MDMS regardless of the default access state or whether there are any access control entries on an object. As always, appropriate MDMS rights are also checked for each request, and setting **/NOCHECK_ACCESS** does not disabled rights validation.

The default is **/NOCHECK_ACCESS**.

This qualifier requires the right MDMS_SET_RIGHTS.

/DEALLOCATE_STATE=*state*

This attribute stores a text string stating the deallocation state of volumes. Allowable values are FREE and TRANSITION. This state is applied to volumes that are deallocated when no specific state is specified on the deallocate request and the volume record does not have a transition time defined.

/DEFAULT_RIGHTS[=(*right*[,...])]

/NODEFAULT_RIGHTS

The **/DEFAULT_RIGHTS** qualifier sets the low-level rights associated with users with no MDMS rights in their UAF record. If the qualifier is specified with no value, a default set of rights (i.e. no rights) are set. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace rights in the list, rather than adding them by default. Use **/NODEFAULT_RIGHTS** to remove all default rights.

This qualifier requires the right MDMS_SET_RIGHTS.

/DESCRIPTION="*text*"

Comments about the domain. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/MAIL_USERS=(*user*[,...])

/NOMAIL_USERS

This attribute is a list of OpenVMS system users who should receive notification when volumes are deallocated. The default is user SYSTEM.

/MAXIMUM_SCRATCH_TIME=*delta-time*

The maximum scratch time is an OpenVMS delta time that specifies the greatest allocation time allowed.

/MEDIA_TYPE=*media-type*

This qualifier specifies the default MDMS media type, which is applied to drives and volumes if they are created without a media type definition.

/OFFSITE_LOCATION=*location***/NOOFFSITE_LOCATION (Default)**

This attribute stores a text string identifying the name of the Location object representing your default off-site storage location. This location is applied to all volumes and magazines that do not have an explicit off-site location specified.

/ONSITE_LOCATION=*location***/NOONSITE_LOCATION (Default)**

This attribute stores a text string identifying the name of the location object representing your default on-site storage location. This location is applied to all volumes and magazines that do not have an explicit on-site location specified.

/OPCOM_CLASSES=(*class*[,...])

Use this attribute to list the OpenVMS OPCOM classes to which MDMS OPCOM messages are directed. These OPCOM classes are applied to all nodes that do not have explicit OPCOM classes specified. The following classes are supported:

CARDS	NETWORK	OPER6	OPER12
CENTRAL	OPER1	OPER7	PRINTER
CLUSTER	OPER2	OPER8	REPLY
DEVICES	OPER3	OPER9	SECURITY
DISKS	OPER4	OPER10	SOFTWARE
LICENSE	OPER5	OPER11	TAPES

/OPERATOR_RIGHTS[=(*right*[,...])**/NOOPERATOR_RIGHTS**

The **/OPERATOR_RIGHTS** qualifier sets the low-level rights associated with the high-level right MDMS_OPERATOR. If the qualifier is specified with no value, a default set of rights is set. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace rights in the list, rather than adding them by default. Use **/NOOPERATOR_RIGHTS** to remove all rights.

This qualifier requires the right MDMS_SET_RIGHTS.

/PROTECTION=*protection*

Assign the default volume protection for all volumes with this attribute. Use the standard OpenVMS protection format. This protection is applied to all volumes that do not have an explicit protection specified.

/RELAXED_ACCESS (Default)**/NORELAXED_ACCESS**

The domain **/ [NO] RELAXED_ACCESS** qualifier determines the default user access to an object if no access control entries exist for the object. If at least one access control entry exists for the object, then the default access is not used and access is controlled by the entries.

If **/RELAXED_ACCESS** is set, then all users have access to the object (provided that the object contains no access control entries). This option allows for a more liberal usage of access control: only use it on objects requiring it, and allow users access to all other objects. This is designed for normal environments, and those whose domain-scope is centrally organized.

If **/NORELAXED_ACCESS** is set, then no users have access to the object (provided that the object contains no access control entries), with the exception of:

- The owner of the object, who always has full access
- Users with domain access control have the same level of access on all objects as they do the domain
- The "Last Updated By" user of the domain has full access to the domain and all objects

/NORELAXED_ACCESS is a more conservative approach to access control. It does not allow anyone access to an object unless they have specifically been granted access with an access control entry. This is designed for secure environments, or those whose domain-scope crosses organizational boundaries.

Note that access control is an object-specific method of controlling access. It is in addition to the normal MDMS rights required to perform operations. Having access control without the associated rights is not enough to operate on an object—both are required.

The default is **/RELAXED_ACCESS**.

This qualifier requires the right MDMS_SET_RIGHTS.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/REQUEST_ID=number

Changes the request identifier for the next MDMS request in the domain.

/SCHEDULER_TYPE

When using the integrated ABS and MDMS scheduler functions, the **/SCHEDULER_TYPE** qualifier defines the type of scheduler used for save and restore operations. Select from one of the following options:

- **INTERNAL** (Default) – MDMS internal scheduler using OpenVMS batch queues. This is the default option, and is recommended for normal backup and restore scenarios.
- **EXTERNAL** – MDMS scheduler access via command procedure. This provides more flexibility on how saves and restores are submitted to the batch queues.
- **SCHEDULER** – External scheduler access via command procedures. This is the required option when using an external scheduler product to schedule saves and restores.

The default is INTERNAL.

/SCRATCH_TIME=delta-time

Assign the default scratch date to volumes by applying the delta time specified with attribute to the allocation date of the volume. This scratch time is applied to all volumes that do not have an explicit scratch date defined.

/SYSPRV (Default)

/NOSYSPRV

The default **/SYSPRV** qualifier enables user with OpenVMS privilege SYSPRV the low-level right MDMS_ALL_RIGHTS, which allows all operations. The **/NOSYSPRV** qualifier does not give users with SYSPRV any additional MDMS rights.

If you set the domain to **/NOSYSPRV**, it is highly recommended that you grant MDMS_ALL_RIGHTS in all accounts that require unconditional access to MDMS. This includes the system administrators' accounts and specifically the SYSTEM account.

The MDMS IVP requires unconditional access from the SYSTEM account during installation using either SYSPRV (if enabled) or MDMS_ALL_RIGHTS.

The default is SYSPRV.

/TRANSITION_TIME=delta-time

Defines the default transition time to be applied to volumes that do not have an explicit transition time defined.

/USER_RIGHTS[=(right[,...])

/NOUSER_RIGHTS

The **/USER_RIGHTS** qualifier sets the low-level rights associated with the high-level right MDMS_USER. If the qualifier is specified with no value, a default set of rights is set. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace rights in the list, rather than adding them by default. Use **/NOUSER_RIGHTS** to remove all rights.

This qualifier requires the right MDMS_SET_RIGHTS.

Examples

The following command sets the name of the domain off-site location to XCYX:

```
$ MDMS SET DOMAIN /OFFSITE_LOCATION=XCYX
```

The following command specifies that OpenVMS Cluster users Norton and Crandle are to be the only users notified on the mail distribution. They will be notified when volumes are deallocated:

```
$ MDMS SET DOMAIN /MAIL_USERS=(NORTON, CRANDLE) /REPLACE
```

The following command prevents the further display of OPCOM messages on terminals enabled for the OPER4 OPCOM class:

```
$ MDMS SET DOMAIN /OPCOM_CLASSES=(OPER4) /REMOVE
```

The following command adds the low-level rights MDMS_SET_ALL and MDMS_SET_PROTECTED to the high-level right MDMS_OPERATOR:

```
$ MDMS SET DOMAIN /OPERATOR_RIGHTS=(MDMS_SET_ALL, MDMS_SET_PROTECTED)
```

MDMS SET DRIVE

MDMS SET DRIVE — Modifies a drive definition in the MDMS database.

Format

```
MDMS SET DRIVE [drive-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

drive-name

Specifies the name of the drive. Specify a drive name or a logical name.

The maximum length of the drive name or logical name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of drive names may be entered.

Description

The **MDMS SET DRIVE** command modifies a drive definition in the MDMS database.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The **/JUKEBOX** and **/STACKER** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS=keyword

This qualifier defines the type of access to the drive, which can be one of the following keywords:

ALL (Default)	Supports local node/cluster/fibre access and remote (RDF) access
LOCAL	Supports local node/cluster/fibre access only
REMOTE	Supports remote (RDF) access only

Access to drives is restricted on allocate requests; for example, it is not possible to allocate a drive designated as local access remotely using RDF. However, with the proper rights, it is possible to issue other MDMS commands (such as **LOAD**) both locally and remotely.

The default is **ALL**.

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- **SET** – Allows setting the attributes of the object
- **SHOW** – Allows showing the object
- **DELETE** – Allows deleting the object
- **READ** – Allows restoring data using the object
- **WRITE** – Allows saving data using the object
- **EXECUTE** – Allows execution operations using the object
- **CONTROL** – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list.

This is the default when specifying list items.

/AUTOMATIC_REPLY (Default)
/NOAUTOMATIC_REPLY

Specifies that MDMS automatically replies to all OPCOM messages that can be polled for completion on requests for this particular drive.

The default is **/AUTOMATIC_REPLY**.

/DESCRIPTION="*text*"

Comments about the drive. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/DEVICE=*vms-device-name*

Specifies the OpenVMS device name of the drive. This qualifier is required if the OpenVMS device name is different from the drive name. Do not specify a node specification (NODE::*name*) in the *vms-device-name* parameter. Instead, use the **/NODES** or **/GROUPS** qualifier.

The default device name is the drive name.

/DISABLED

Places the drive in the disabled state. This prevents the drive from being selected and allocated for use. This takes effect immediately. However, if the drive is already in use, operations on that drive will continue until the drive is deallocated.

The default is **/ENABLED**.

/DRIVE_NUMBER=*number*

This qualifier defines the drive number for robot commands if the drive is in a jukebox. This qualifier must be specified for multi-drive, MRD-controlled jukeboxes.

The default is drive number 0.

/ENABLED (Default)

Places the drive in the enabled state. This allows the drive to be selected and allocated for use. This takes effect immediately.

The default is **/ENABLED**.

/GROUPS=(*group-name*[,...])
/NOGROUPS

Specifies the names of groups of nodes that share common access to this device. Usually, only one group is specified. If neither the **/NODES** nor **/GROUPS** qualifier is specified, then the node from which the command was issued is used as the node name. Groups can also be specified with the **/NODES** qualifier.

/JUKEBOX=*jukebox-name*

If the drive is in a jukebox, this qualifier specifies the jukebox name.

/MEDIA_TYPE=(*media-type*[,...])
/NOMEDIA_TYPE

Specifies one or more distinct media types that the drive can support for read-write access. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace objects in the list, rather than adding them by default. The **/NOMEDIA_TYPES** qualifier removes all media types. When a drive is created with no media types, the default media type from the domain is used.

/NODES=(*node-name*[,...])
/NONODES

Specifies one or more distinct nodes that have direct access to the drive. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace objects in the list, rather than adding them by

default. The **/NONODE** qualifier removes all nodes. If neither the **/NODES** nor **/GROUPS** qualifier is specified, then the node from which the command was issued is used as the node name.

/OWNER_NAME=node::username
/OWNER_NAME=group::username
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/READONLY=(media-type[,...])
/NOREADONLY

Specifies one or more distinct media types that the drive can support for read-only access. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace objects in the list, rather than adding them by default. The **/NOREADONLY** qualifier removes all read-only media types.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/SHARED
/NOSHARED (Default)

The **/SHARED** qualifier defines that the drive may be used by non-MDMS clients and that the drive is only partially managed. If the drive is set to the default **/NOSHARED**, the MDMS server allocates the drive at all times it is not used by an application or user. Setting the drive to **/SHARED** clears this allocation.

The default is **/NOSHARED**.

/STACKER
/NOSTACKER (Default)

The **/STACKER** qualifier indicates that the drive is to be treated as a stacker gravity loader. The default **/NOSTACKER** indicates that the drive is to be treated as either a standalone drive or a robotically-controlled jukebox, as appropriate.

The default is **/NOSTACKER**.

/STATE=state

This is a protected field that should be modified only to recover on error. Use the **LOAD** and **UNLOAD** commands to manipulate the state field under normal operation. The **/STATE** qualifier sets the current drive state. The valid keywords are Empty, Full, Loading, and Unloading.

This qualifier requires the right MDMS_SET_PROTECTED.

Examples

The following command adds nodes JOHN and PAUL for direct access to the drive with drive name or logical name of FRED:

```
$ MDMS SET DRIVE FRED /NODES=(JOHN, PAUL)
```

MDMS SET ENVIRONMENT

MDMS SET ENVIRONMENT — Modifies an existing environment definition in the MDMS database. Environments were formerly known as execution environments or environment policies. Environment is the new common name.

Format

```
MDMS SET ENVIRONMENT [environment-name] [, ...]
```

Equivalent ABS Command: **SET ENVIRONMENT**

Parameters

environment-name

Specifies the name of the environment.

The maximum length of the environment name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of environment names may be entered.

Description

The **MDMS SET ENVIRONMENT** command modifies an existing environment definition in the MDMS database. An environment describes how data is saved and restored during execution.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

This command is restricted to nodes with an ABS license. The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ACTION=*keyword*

The **/ACTION** qualifier specifies an operation to perform on files or data being saved using this environment. Select one of the following options:

RECORD_DATE	Modifies the BACKUP date on files being saved (VMS_FILES only)
NO_CHANGE	Does not perform any operation on the data being saved
DELETE_OBJECT	Deletes the data being saved—this is typically used for long-term archive operations

If not specified, the default is RECORD_DATE.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list.

This is the default when specifying list items.

/ASSIST (Default)**/NOASSIST**

The **/ASSIST** qualifier specifies whether saves and restores using this environment utilize operator assistance (using OPCOM) when needed to continue the operation.

If **/ASSIST** is specified and operator intervention is needed an OPCOM message will be output. If **/NOASSIST** is specified and an operator intervention is needed, the operation will fail and the next operation will be performed (if possible).

The default is **/ASSIST**.

/DATA_SAFETY=(keyword[,...])**/NODATA_SAFETY**

The **/DATA_SAFETY** qualifier specifies one or more options that the backup agent uses to ensure the integrity of the data being saved. The qualifier is applicable to archive type TAPE and when using OpenVMS Backup as the backup agent. Select one or more of the following options:

CRC	Performs a Cyclic Redundancy check and writes it for each data block on a tape volume
FULL_VERIFY	Rereads all saved data and compares to what is on disk
XOR	If the CRC check detects a bad block during a restore operation, the XOR mechanism allows recovery of the block.

By default, data safety options are added. However, you can use the **/REMOVE** qualifier to remove any option, and the **/REPLACE** qualifier to replace the option list. If you specify **/NODATA_SAFETY**, then all options are removed.

If not specified, the default options are: (CRC, FULL_VERIFY, XOR).

/DESCRIPTION="text"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/DRIVE_COUNT=number

The **/DRIVE_COUNT** qualifier specifies the number of drives to use for each save or restore using this environment. If there are at least as many drives available as the drive count, that number of drives are allocated for each save and restore request. If not, a reduced number of drives are allocated.

The default and highly recommended value is 1. The number of drives may range from 1 to 32.

/EPILOGUE="string"**/NOEPILOGUE (Default)**

The **/EPILOGUE** qualifier specifies a string to execute after an entire save or restore operations completes. The string may be up to 80 characters and must be enclosed in quotes.

By default, no epilogue command is executed.

/INTERVAL=*delta-time*

The **/INTERVAL** qualifier specifies the time between retries if a save or restore operation fails. This is used in association with the **/RETRY_LIMIT** qualifier. The interval range must be between 1–60 minutes.

If **/RETRY_LIMIT** is greater than zero, then the default interval applied is 00:15:00 (15 minutes).

/LISTING_OPTION=*keyword*

The **/LISTING_OPTION** qualifier specifies the level of detail in the list file for a save or restore operation. Select one of the following options:

NONE (Default)	No listing file is generated
BRIEF	Generates a brief listing file
FULL	Generates a full listing file including full file names and file information provided by the backup agent

If not specified, the default is NONE.

/LOCK**/NOLOCK (Default)**

The **/LOCK** qualifier specifies the action to take when data usage conflicts occur. If **/LOCK** is specified, data is saved even if other applications have the data locked for write access. If **/NOLOCK** is specified, the data is not saved if other applications have the data locked for write access.

The default is **/NOLOCK**.

/NOTIFICATION=(*option*[,...])**/NONOTIFICATION**

The **/NOTIFICATION** qualifier specifies options to use for notifying mail user or OPCOM classes when using this environment. Select one or more of the following options:

- **MAIL=(*username*[,...])** – Specifies one or more mail users to receive mail on completion of an operation. The username may include a node name (*node::username*) or group name (*group::username*). By default the mail is sent to the user on the node on which the request executes.
- **OPCOM=(*class*[,...])** – Specifies one or more OPCOM classes to receive an OPCOM on completion of an operation. The class names must be valid OpenVMS OPCOM class names (TAPES etc).
- **TYPE=*keyword*** – Specifies the level of detail included in the notification. This keyword is applicable to MAIL users only. Select one of the following options:
 - BRIEF
 - FULL
 - NORMAL
- **WHEN=(*option*[,...])** – Specifies when the notification is sent and under what circumstances. Select one or more of the following options:

- COMPLETE – Sends notification at the completion of a request with any status
- ERROR – Sends notification at the completion of a request with error or fatal status
- FATAL – Sends notification at the completion of a request with fatal status
- START – Sends notification at the start of a request
- WARNING – Sends notification at the completion of a request with warning, error or fatal status

Only one notification entry can be entered per command. Use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to add, remove or replace mail users, OPCOM classes or when definitions. The usernames in an entry must match exactly to successfully modify or remove an entry.

If not specified, the default is:

(OPCOM=TAPE, TYPE=BRIEF, WHEN=FATAL)

/OWNER_NAME=node::username
/OWNER_NAME=group::username
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/PROFILE=(USER=username, option[,...])

The **/PROFILE** specifies the execution user context when running saves or restore in this environment.

A profile must contain a username, and one or more of the following options:

- **CLUSTER=cluster-name** – A name of a cluster (group). This is used together with the username to identify the user account on the cluster. Specify "*" for all clusters.
- **NODE=node-name** – A name of a node. This is used together with the username to identify the user account on the node. Specify "*" for all nodes.
- **PRIVILEGES=(privileges[,...])** – A list of OpenVMS privileges. If specified, this overrides the default privileges obtained from the user's UAF record on the node or cluster.
- **RIGHTS=(right[,...])** – A list of rights names. If specified, this overrides the default rights obtained from the user's UAF record on the node or cluster.

If the username is not specified, the existing username in the profile is assumed. The username must not contain a node or cluster name, and must be a valid OpenVMS user on the node or cluster specified.

If the profile is not specified, the following defaults are applied:

```
USER=ABS
CLUSTER=*
NODE=*
PRIVILEGES=NONE (will use ABS privileges on execution node)
RIGHTS=NONE (will use ABS rights on execution node)
```

If just a username is provided, the default node is the node in which the command is entered. The privileges and rights are obtained from the UAF file for the specified user on the specified node (or the command node if none).

User name can also be specified as a special keyword, <REQUESTER>. This means that the user profile in the associated save or restore request is used. This is the normal situation for user backups. The other profile fields are not used in this case.

/PROLOGUE="string"
/NOPROLOGUE (Default)

The **/PROLOGUE** qualifier specifies a string to execute before an entire save or restore operations starts. The string may be up to 80 characters and must be enclosed in quotes.

By default, no prologue command is executed.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/RETRY_LIMIT=number
/NORETRY_LIMIT

The **/RETRY_LIMIT** qualifier specifies the number of retries performed if a save or restore operation fails. If you specify a retry limit of 0, no retries are attempted. **/NORETRY_LIMIT** means that there is no limit on the number of retries.

If not specified, the default value is 3.

Examples

The following command modifies the environment `USER_SAVE_ENV`, specifying an action to not modify files on saves, changing the user profile to ABS on any node or cluster, and replacing notification for OPCOM classes `CENTRAL` and `TAPES` to use a `WHEN of ERROR`:

```
$ MDMS SET ENVIRONMENT USER_SAVE_ENV -
_$ /ACTION=NO_CHANGE -
_$ /PROFILE=(USER=ABS, CLUSTER=*, NODE=*) -
```

```
_$_ /NOTIFICATION=(OPCOM=(CENTRAL,TAPES),WHEN=ERROR)/REPLACE
```

MDMS SET GROUP

MDMS SET GROUP — Modifies a group definition in the MDMS database.

Format

```
MDMS SET GROUP [group-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

group-name

Specifies the name of the group.

The maximum length of the drive name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of group names may be entered.

Description

The **MDMS SET GROUP** command modifies a group definition in the MDMS database.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

```
/ACCESS_CONTROL=(USERNAME=user-id, ALLOW=(option[,...]))  
/NOACCESS_CONTROL (Default)
```

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object

- **WRITE** – Allows saving data using the object
- **EXECUTE** – Allows execution operations using the object
- **CONTROL** – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/DESCRIPTION="text"

Comments about the group. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/NODES=(node-name[,...])

/NONODES

Specifies one or more nodes that comprise the group. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace objects in the list, rather than adding them by default. The **/NONODES** qualifier removes all nodes.

/OWNER_NAME=node::username

/OWNER_NAME=group::username

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

Examples

The following command removes node `TOOKUS` from the group `HOSER`:

```
$ MDMS SET GROUP HOSER /NODE=TOOKUS /REMOVE
```

MDMS SET JUKEBOX

MDMS SET JUKEBOX — Modifies a jukebox definition in the MDMS database.

Format

```
MDMS SET JUKEBOX [jukebox-name] [, ...]
```

Equivalent **STORAGE** Command: None.

Parameters

jukebox-name

Specifies the name of the jukebox.

The maximum length of the jukebox name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of jukebox names may be entered.

Description

The **MDMS SET JUKEBOX** command modifies a jukebox definition in the MDMS database.

Privileges

The request requires `MDMS_SET_ALL`.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS=keyword

This qualifier defines the type of access to the jukebox, which can be one of the following keywords:

ALL (Default)	Supports local node/cluster/fibre access and remote (RDF) access

LOCAL	Supports local node/cluster/fibre access only
REMOTE	Supports remote (RDF) access only

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ACS=*acs-id*

The **/ACS** qualifier specifies the Automated Cartridge System (ACS) identifier. The default value is zero. Each MDMS jukebox maps to one Library Storage Module (LSM), and requires the specification of the library, ACS and LSM identifiers. Valid for DCSC-controlled jukeboxes only.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/AUTOMATIC_REPLY (Default)
/NOAUTOMATIC_REPLY

Specifies that MDMS automatically replies to all OPCOM messages that can be polled for completion on requests for this particular jukebox.

/CAP_SIZE=(*number*[,...])

For DCSC-controlled jukeboxes equipped with Cartridge Access Points (CAPs), this attribute specifies the number of cells for each CAP. The first number is the size for CAP 0, the next for CAP 1 etc. If a size is not specified for a CAP, a default value of 40 is used. Specifying the CAP size

optimizes the movement of volumes to and from the jukebox by filling the CAP to capacity for each move operation. When specifying CAP sizes, the specified numbers always replace any previous sizes in the database. Valid for DCSC-controlled jukeboxes only.

/CONTROL=keyword

This qualifier specifies the robot control facility used to control the jukebox's robot. The valid keywords are:

MRD (Default)	The robot is controlled with the MRD facility
DCSC	The jukebox is a silo controlled by the DCSC facility (for StorageTek(R) Silos only)

/DESCRIPTION="text"

Defines comments about the object in the record. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the existing description, specify "".

/DISABLED

Places the jukebox in the disabled state. This prevents all drives in the jukebox from being selected and allocated for use. This takes effect immediately. However, any drives in use will continue to be used until they are deselected.

/ENABLED (Default)

Places the drive in the enabled state. This allows drives in the jukebox to be selected and allocated for use. This takes effect immediately. This is the default.

/GROUPS=(group-name[,...])

/NOGROUPS

Specifies the names of groups of nodes that share common access to this device. Normally, only one group is specified. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace objects in the list, rather than adding them by default. The **/NOGROUPS** qualifier removes all groups.

/LIBRARY=library-id

This qualifier specifies the library identifier for a silo. Valid values are 1,2,3,4, and the default is 1 when the jukebox is controlled by DCSC and 0 (not applicable) when controlled by MRD. Each MDMS jukebox maps to one Library Storage Module (LSM), and requires specification of the library, ACS and LSM identifiers.

/LOCATION=location-name

/NOLOCATION

This qualifier specifies the location of the jukebox, which is used when moving volumes into and out of the jukebox. If not specified, or if **/NOLOCATION** is specified, the default on-site location from the domain record is used as the jukebox location.

/LSM=lsn-id

The **/LSM** qualifier specifies the Library Storage Module (LSM) identifier. The default value is zero. Each MDMS jukebox maps to one Library Storage Module (LSM), and requires the specification of the library, ACS and LSM identifiers. Valid for DCSC-controlled jukeboxes only.

/NODES=(*node-name*[,...])
/NONODES

Specifies one or more nodes that can directly access the jukebox. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace objects in the list, rather than adding them by default. The **/NONODES** qualifier removes all nodes.

/OWNER_NAME=*node::username*
/OWNER_NAME=*group::username*
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/ROBOT=*robot-name*
/NOROBOT

The **/ROBOT** qualifier defines the OpenVMS device name of the robot. Required for, and applicable to MRD-controlled jukeboxes only.

Do not specify a node name in the robot name.

/SLOT_COUNT=*number*

The **/SLOT_COUNT** qualifier specifies the total number of slots in the entire jukebox. For any MRD jukebox, either the slot count or topology must be specified. Valid for MRD-controlled jukeboxes only.

/STATE=*keyword*

This is a protected field that should be modified only to recover on error. Use the **LOAD**, **UNLOAD**, or **MOVE** commands to manipulate the state field under normal operation. The **/STATE** qualifier specifies the usage state of the jukebox. The keyword values are:

Available	The jukebox is available for use
In_use	The jukebox is in use

This qualifier requires the right MDMS_SET_PROTECTED.

/THRESHOLD=*number*

Specifies that an OPCOM message is output when the number of free volumes in the jukebox falls below the specified number. The default value is zero, which disables the feature. The OPCOM message is output on a node that can directly access the jukebox.

/TOPOLOGY=(TOWERS=(*number*[,...]), FACES=(*number*[,...]), LEVELS=(*number*[,...]), SLOTS=(*number*[,...]))

Specifies topology of jukebox, when a TL820-class jukebox is being used as a magazine. Valid for MRD-controlled jukeboxes only. The topology specification allows OPCOM messages to move magazines to be specified with TOWER, FACE, LEVEL rather than slot range. The specification of topology is optional. For each tower in the configuration, a corresponding entry must also be placed in FACES LEVELS and SLOTS that reflects the configuration of that tower.

The tower numbers start at zero, and additional towers must be the next number in sequence (i.e. 0,1,2 etc). Other specifications are absolute counts of the entity being specified for each tower (i.e. the total number of faces, levels and slots in each tower).

For example, for a `three-tower` jukebox, each tower having `eight` faces, the first tower having `two` levels and the other two towers having `three` levels, and support of `11-slot` bin-packs, the topology specification would be:

```
/TOPOLOGY=(TOWERS=(0,1,2), /FACES=(8,8,8), LEVELS=(2,3,3),  
SLOTS=(11,11,11))
```

/USAGE=[NO]MAGAZINE

The **/USAGE=MAGAZINE** qualifier specifies that the jukebox is configured for magazines, and that the movement of volumes may be performed using magazines. The **/USAGE=NOMAGAZINE** qualifier does not support magazine use. The default is NOMAGAZINE.

You must specify **/USAGE=MAGAZINE** when defining the **/TOPOLOGY** attribute. Note that you can use the jukebox for non-magazine moves even when the usage is magazine, but the reverse is not true. Valid for MRD-controlled jukeboxes only.

Examples

The following command modifies the description for jukebox JUKE_1:

```
$ MDMS SET JUKEBOX JUKE_1 /DESCRIPTION="Design Data Backup"
```

The following command modifies the jukebox JUKE_2 to support magazines, and slots 0-100:

```
$ MDMS SET JUKEBOX JUKE_2 /USAGE=MAGAZINE /SLOTS=(0-100)
```

MDMS SET LOCATION

MDMS SET LOCATION — Modifies a location definition in the MDMS database.

Format

```
MDMS SET LOCATION [location] [,...]
```

Equivalent STORAGE Command: None.

Parameters

location

Specifies the name of the location.

The maximum length of the location is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of locations may be entered.

Description

The **MDMS SET LOCATION** command modifies a location definition in the MDMS database.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

None.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/DESCRIPTION="text"

Defines comments about the object in the record. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the existing description, specify "".

/LOCATION=location

/NOLOCATION (Default)

The **/LOCATION** qualifier allows you to specify a parent location, thus creating a location hierarchy. If there is no parent location, specify **/NOLOCATION**. Use parent locations to allow selection of volumes or drives in compatible locations. One location is compatible with another if it has a common parent location in the hierarchy. If you do not wish to utilize the compatible location feature, do not specify parent locations. Locations with common parents are most useful where the parents and siblings are in close proximity to one another (e.g. rooms 101 and 102, with parent location floor 1), and selection of volumes or drives from any of the locations is desired. Do not use parent locations across larger distances.

/OWNER_NAME=node::username

/OWNER_NAME=group::username

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/SPACES=(range)

/NOSPACES

The **/SPACES** qualifier defines individual spaces for volumes or magazines at the location. Spaces are alphanumeric strings of up to 8 characters. The spaces can be specified as a range—only a single range is supported. The **/NOSPACES** qualifier removes all spaces.

Examples

The following command modifies the location called `SHELF_100` and supports spaces 300–500:

```
$ MDMS SET LOCATION SHELF_100 /SPACES=(300-500)
```

MDMS SET MAGAZINE

MDMS SET MAGAZINE — Modifies a magazine definition in the MDMS database.

Format

MDMS SET MAGAZINE [*magazine-name*] [, ...]

Equivalent STORAGE Command: None.

Parameters

magazine-name

Specifies the name of the magazine.

The maximum length of the magazine name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of magazine names may be entered.

Description

The **MDMS SET MAGAZINE** command modifies a magazine definition in the MDMS database.

Privileges

The request requires MDMS_SET_ALL.

The **/JUKEBOX**, **/PLACEMENT**, **/POSITION**, and **/START_SLOT** qualifiers also require MDMS_SET_PROTECTED.

Restrictions

None.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/DESCRIPTION="text"

Defines comments about the object in the record. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the existing description, specify "".

/JUKEBOX=jukebox-name

This is a protected field that should only be modified to recover on error. Use the **MOVE MAGAZINE** command to set up the jukebox name under normal operations. This qualifier specifies the name of the jukebox in which the magazine resides. The maximum length of the jukebox name is 31 characters.

This qualifier requires the right MDMS_SET_PROTECTED.

/OFFSITE=(LOCATION=location)[,[NO]DATE[=date]] **/NOOFFSITE**

This qualifier specifies the date that the magazine is to be taken off site and the off-site location. The location field is required when using the **/OFFSITE** qualifier if no location has been previously specified. The LOCATION keyword cannot be negated and if specified must point to a valid location object. Specify a VMS absolute or delta time in the date field. The NODATE keyword may be used to remove the date. This has the effect of disabling the schedule for the magazine, while retaining the location. To clear the off-site date, specify **/NOOFFSITE**.

The default off-site location is that stored in the domain object.

/ONSITE=(LOCATION=location)[,[NO]DATE[=date]] **/NOONSITE**

This qualifier specifies the date that the magazine is to be brought back on site and the on-site location. The location field is required when using the **/ONSITE** qualifier if no location has been previously specified. The LOCATION keyword cannot be negated and if specified must point to a valid location object. Specify a VMS absolute or delta time in the date field. The NODATE keyword may be used to remove the date. This has the effect of disabling the schedule for the magazine, while retaining the location. To clear the on-site date, specify **/NOONSITE**.

The default on-site location is that stored in the domain object.

/OWNER_NAME=node::username **/OWNER_NAME=group::username** **/NOOWNER_NAME**

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/PLACEMENT=keyword

This is a protected field that should only be modified to recover on error. Use the **MOVE MAGAZINE** command to set up the placement under normal operation. This qualifier defines the current placement of the magazine. Select one of following options:

JUKEBOX
OFFSITE
ONSITE
MOVING

This qualifier requires the right MDMS_SET_PROTECTED.

/POSITION=position=(tower,face,level)

This is a protected field that should only be modified to recover on error. Use the **MOVE MAGAZINE** command to set up the position under normal operation. The **/POSITION** qualifier specifies the position in the jukebox that the magazine resides.

The tower, face and level specification represent the relative number of the tower, face and level, starting from 0. So for absolute jukebox slot of zero, the corresponding position is (0,0,0). The next position in the jukebox would be (0,0,1) and so on, according to the topology defined for the jukebox.

This qualifier requires the right MDMS_SET_PROTECTED.

/SLOT_COUNT=number

The **/SLOT_COUNT** qualifier specifies the number of slots in a magazine to store volumes.

/SPACES=(range)
/NOSPACES

This qualifier specifies the spaces in a location in which the magazine is stored when not in a jukebox. Spaces are alphanumeric strings of up to 8 characters. The **/NOSPACES** qualifier removes all spaces.

/START_SLOT=(number)

This is a protected field that should only be modified to recover on error. Use the **MOVE MAGAZINE** command to set up the start slot under normal operation. This qualifier specifies the starting jukebox slot when the magazine is placed in a jukebox.

This qualifier requires the right MDMS_SET_PROTECTED.

Examples

The following command modifies the non-jukebox location of magazine MYMAG01 to space S4 in SHELF_20:

```
$ MDMS SET MAGAZINE MYMAG01 /ONSITE=(LOCATION=SHELF_20) /SPACES=S4
```

MDMS SET MEDIA_TYPE

MDMS SET MEDIA_TYPE — Modifies a media type definition in the MDMS database.

Format

```
MDMS SET MEDIA_TYPE [media-type] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

media-type

Specifies the name of the media type.

The maximum length of the media type name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of media types may be entered.

Description

The **MDMS SET MEDIA_TYPE** command modifies a media type definition in the MDMS database. A media type definition consists of a density, compaction option, and capacity (if applicable).

Privileges

The request requires MDMS_SET_ALL.

Restrictions

None.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object

- **DELETE** – Allows deleting the object
- **READ** – Allows restoring data using the object
- **WRITE** – Allows saving data using the object
- **EXECUTE** – Allows execution operations using the object
- **CONTROL** – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/CAPACITY=number

The **/CAPACITY** qualifier specifies the capacity in megabytes of the tape. This is used by some MDMS clients to estimate end-of-tape conditions. By default, capacity is set to zero.

/COMPACTION (Default)

/NOCOMPACTION

The **/COMPACTION** qualifier specifies that the media type should use compaction when writing to tape. This is the default. If you do not wish to use compaction, then specify **/NOCOMPACTION**.

/DENSITY=density

Specifies a density string between 1 and 31 characters in length that the media type supports. Note that the COMP keyword for compaction should be specified in the **/COMPACTION** attribute, not density.

/DESCRIPTION="text"

Defines comments about the media type. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description is can range from 0 to 255 characters.

Specify "" to clear the description.

/LENGTH=length

The **/LENGTH** qualifier specifies the length of a 9-track magnetic tape, and is expressed in feet. By default, length is set to zero.

/OWNER_NAME=node::username

/OWNER_NAME=group::username

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

Examples

The following command modifies a media type called TAPE_9T to use density of 1600 and length of 3600:

```
$ MDMS SET MEDIA_TYPE TAPE_9T /DENSITY=1600 /LENGTH=3600
```

MDMS SET NODE

MDMS SET NODE — Modifies a node definition in the MDMS database.

Format

```
MDMS SET NODE [node-name] [, ...]
```

Equivalent STORAGE Command: None.

Parameters

node-name

Specifies the name of the node. Do not append colons to the node name. The node name should be the DECnet Phase IV node name (i.e. logical SYS\$NODE) if DECnet Phase IV is supported on the node. Otherwise, it should be a unique name chosen by the MDMS administrator.

If DECnet-Plus and/or TCP/IP are supported, the appropriate fullnames should be stored as attributes of the node. Do not use the node name to specify fullnames.

The maximum length of the node name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

Description

The **MDMS SET NODE** command modifies a node definition in the MDMS database.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/DATABASE_SERVER (Default)

/NODATABASE_SERVER

The **/DATABASE_SERVER** qualifier means the node can be a database server, supporting failover operations. To be a database server, the node must have direct access to the MDMS database files.

/DECNET_PLUS_FULLNAME=*node-fullname*

This qualifier allows you to specify the DECnet-Plus fullname for a node. The fullname may be up to 255 characters. If this node has a DECnet-Plus name defined by the logical name "SYS\$NODE_FULLNAME", then the DECNET_PLUS_FULLNAME has to be defined for this node and has to exactly match the DECnet-Plus name. The DECNET_PLUS_FULLNAME has to be defined in order for this node to be fully enabled when the DECnet transport has been enabled and DECnet-Plus is running on the system. The fullname can be specified in upper or lower case.

/DESCRIPTION=*text*

Defines comments about the node. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. Specify "" to clear the description.

/DISABLED

Places the node in the disabled state. This prevents the node from participating in the MDMS domain as either a server or a client. This takes effect immediately.

/ENABLED (Default)

Places the node in the enabled state. This allows the node to participate in MDMS operations. This takes effect immediately. This is the default.

/LOCATION=*location***/NOLOCATION**

The **/LOCATION** qualifier specifies the location of the node, which is used when allocating volumes and drives. If not specified, or if **/NOLOCATION** is specified, the default on-site location from the domain record is used as the node location.

/OPCOM=(*class*[,...])**/NOOPCOM**

The **/OPCOM** qualifier adds the specified classes used for notifying operators. All OPCOM for devices on the node are sent to all specified classes on the node. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace classes in the list, rather than adding them by default. Specify **/NOOPCOM** to disable OPCOM notification. By default, the node acquires OPCOM classes from the domain record.

The following classes are valid:

CARDS	NETWORK	OPER6	OPER12
CENTRAL	OPER1	OPER7	PRINTER
CLUSTER	OPER2	OPER8	REPLY
DEVICES	OPER3	OPER9	SECURITY
DISKS	OPER4	OPER10	SOFTWARE
LICENSE	OPER5	OPER11	TAPES

/OWNER_NAME=*node::username***/OWNER_NAME=*group::username*****/NOOWNER_NAME**

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an

owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/TCPIP_FULLNAME=*node-fullname[:low-port-high-port]*

This qualifier allows you to specify the TCP/IP full name for a node. The full name may be up to 255 characters. If this node has a TCP/IP name defined by the logical name *"*INET_HOST"*, then the **TCPIP_FULLNAME** has to be defined and has to exactly match the full IP name as *"<INET_HOST>.<INET_DOMAIN>"*. For *INET_DOMAIN*, see the logical name *"*INET_DOMAIN"*. The **TCPIP_FULLNAME** has to be defined in order for this node to be fully enabled when the TCP/IP transport has been enabled. The fullname can be specified in upper or lower case.

The *low-port* and *high-port* numbers specify the range of TCP/IP port numbers used by the server to listen for incoming requests. The default is 2501-2510. If this conflicts with other applications, a new range above 1023 can be specified. The range should contain at least 10 port numbers for the MDMS server to select one at a time.

Note that the MDMS GUI requires TCP/IP running on all GUI nodes, and on the MDMS server nodes to which the GUI may connect.

/TRANSPORT=(*keyword*[,...])

Specifies the transport protocols to be used when communicating with other MDMS facilities on other nodes, as a prioritized ordered list. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace objects in the list, rather than adding them by default.

Enter one or more of:

- **DECNET** – Listen to incoming requests from other MDMS servers on DECnet Phase IV and DECnet-Plus
- **TCPIP** – Listen to incoming requests from other MDMS servers on TCP/IP

Setting a new transport will automatically start the listener for this transport on the database server node. Likewise, removing a transport will take place within 10 seconds on the database server node. For client nodes, transport changes will take place the next time network connections time out (usually within 10 minutes). If the change needs to take place immediately, the client node server process must be restarted.

The node name and/or the node full names have to be set accordingly for a transport to work correctly.

Examples

The following command modifies a node definition named `COOKIE` by adding OPCOM classes `TAPES` and `OPER1` to the current list of OPCOM classes:

```
$ MDMS SET NODE COOKIE /OPCOM=(TAPES, OPER1)
```

MDMS SET POOL

MDMS SET POOL — Modifies a pool definition in the MDMS database.

Format

```
MDMS SET POOL [pool-name] [, ...]
```

Equivalent **STORAGE** Command: None.

Parameters

pool-name

Specifies the name of the pool.

The maximum length of the pool name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of pool names may be entered.

Description

The **MDMS SET POOL** command modifies a pool definition in the MDMS database.

Privileges

The request requires `MDMS_SET_ALL`.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- **SET** – Allows setting the attributes of the object
- **SHOW** – Allows showing the object

- **DELETE** – Allows deleting the object
- **READ** – Allows restoring data using the object
- **WRITE** – Allows saving data using the object
- **EXECUTE** – Allows execution operations using the object
- **CONTROL** – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list.

This is the default when specifying list items.

/AUTHORIZED_USERS=(node/group-name::username[,...])
/NOAUTHORIZED_USERS

Specifies one or more distinct users to the pool specified by node or group name and user name. Only authorized or default users can allocate volumes belonging to the pool. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace users in the list, rather than adding them by default.

/DEFAULT_USERS=(node/group-name::username[,...])
/NODEFAULT_USERS

Specifies one or more distinct users to the pool as the users default pool. Only authorized or default users can allocate volumes belonging to the pool. The **/REMOVE** or **/REPLACE** qualifiers can be used to remove or replace users in the list, rather than adding them by default. A particular *node/group-name::username* combination should only be defined with the **/DEFAULT** qualifier for one pool.

/DESCRIPTION="text"

Defines comments about the pool. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. Specify "" to clear the description.

/OWNER_NAME=node::username
/OWNER_NAME=group::username
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/THRESHOLD=number

Specifies that an OPCOM message is output when the number of free volumes in the pool falls below the specified number. The default value is zero, which disables the feature. The OPCOM message is output on the database server node.

Examples

The following command adds authorized users `COOKIE::ABS` and `COOKIE::HSM` to a pool called `TEST_POOL`:

```
$ MDMS SET POOL TEST_POOL/AUTHORIZE=(COOKIE::ABS, COOKIE::HSM)
```

MDMS SET RESTORE

MDMS SET RESTORE — Modifies an existing restore definition in the MDMS database. Restores were formerly known as restore requests or restore transactions. Restore is the new common name.

Format

```
MDMS SET RESTORE [restore-name] [,...]
```

Equivalent ABS Command: **SET RESTORE**

Parameters

restore-name

Specifies the name of the restore.

The maximum length of the restore name is 39 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of restore names may be entered.

Description

The **MDMS SET RESTORE** command modifies an existing restore definition in the MDMS database. Restores are used to copy previously-saved data from an archive back to online disk storage.

The data specifications for restores can be supplied in one of two ways:

- Using the **/INCLUDE**, **/EXCLUDE**, **/DATA_SELECT_TYPE**, and **/SOURCE_NODE** qualifiers. These qualifiers can be used to specify a set of data specifications of a single data select type that can be restored. These are not positional qualifiers. This option is recommended for most restores, and this information is stored in a default selection that is mapped to the restore by MDMS. The default selection name is *restore-name_REST_SEL_DEF*.

See [MDMS CREATE RESTORE](#) for the note on maintaining file directory structure during the Restore operation.

- For more complex data specifications, you can create your own selections and map them to the restore using the **/SELECTIONS** qualifier. This options gives you more granularity in data selection, and allows you to have mixed data select types in a single restore.

Most restores are **ONE_TIME_ONLY** (the default) and do not need to be scheduled. However, restores can also be regularly scheduled in the same way as saves. If you wish to use one of the supplied frequency options, MDMS creates a schedule and initializes it to appropriate values. The schedule for any restore is called *restore-name_REST_SCHED*.

Alternatively, if you wish to define a schedule not handled in the supplied frequencies, you can use one of the following options:

- For scheduler types **INTERNAL** and **EXTERNAL**, you can define **/FREQUENCY=CUSTOM**, and modify the schedule to your own specification.
- For scheduler type **SCHEDULER**, you can define **/FREQUENCY=EXPLICIT**, and define a string to pass the explicit schedule to the scheduler in the qualifier **/EXPLICIT_INTERVAL**.

Privileges

The request requires **MDMS_SET_ALL**.

Restrictions

This command is restricted to nodes with an ABS license.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The **/DELETE_INTERVAL** and **/KEEP** qualifiers are mutually exclusive.

The **/BEFORE_DATE**, **/DATE_ARCHIVE**, and **/SINCE_DATE** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/ARCHIVE=(*archive-name*[,*archive-name*])

The **/ARCHIVE** qualifier allows you to specify one or two archives associated with the restore. The first archive name is applicable to full and selective threads in restores. If specified, the second archive name is associated with incremental threads in restores. If a second name is not specified, incremental threads use the first archive name. For any thread, the catalog name from the first archive is always used. At least one archive name must be supplied for a restore.

If an archive is not specified, the archive "SYSTEM_BACKUPS" is used by default.

/BASE_DATE=*date* **/NOBASE_DATE**

The base date is used as a starting point for restores that have a repeating schedule frequency. Subsequent schedules are applied by adding the multiples of the frequency to the base date. If the restore is not to be repeated, then no base date is needed.

By default, if a base date is not specified but a start date is, the base date is set to the start date. If neither is specified, the base date is not set and the restore is not scheduled for execution.

/BEFORE_DATE=*date*
/NOBEFORE_DATE

You can select files for restore based on the date archived before a certain date. If specified, the restore will apply to the latest date archived before the specified date. **/NOBEFORE_DATE** clears the before date and uses other date criteria to select files, as follows:

- **DATE_ARCHIVED** – The latest file saved on the specified date
- **SINCE_DATE** – The first file saved after the specified date
- **No date** – The latest copy of the file

/CATALOG=(NAME=*catalog-name***, [NO]NODES=(***node***[,...]))**
/NOCATALOG (Default)

The **/CATALOG** qualifier specifies the name of the catalog used by the restore, and a list of nodes and groups that can directly access the catalog. This catalog overrides the catalog defined in the first archive specified in the restore. If you specify **/NOCATALOG**, then the catalog in the first restore archive is used.

Specify a local catalog name representing a catalog object in the MDMS database. With the **NODES** keyword, you may specify nodes, groups or a combination thereof. If you specify **NONODES**, the local execution node on which the restore is performed is used as the catalog node.

The default is **/NOCATALOG**.

/DATA_SELECT_TYPE=*keyword*

The **/DATA_SELECT_TYPE** qualifier specifies the type of data to restore in the default selection. Select one of the following options:

- **VMS_FILES**
- **VMS_SAVESET**

If not specified, MDMS attempts to determine the data select type based on the include file syntax. If MDMS cannot determine this, **VMS_FILES** is used. The data select type specification is stored in the default selection.

/DATE_ARCHIVED=*date*
/NODATE_ARCHIVED

You can select files for restore based on the date archived. If specified, the restore will apply to the latest copy of the data on the date archived. **/NODATE_ARCHIVED** clears the date archived and uses other date criteria to select files, as follows:

- **BEFORE_DATE** – The latest file saved before the specified date
- **SINCE_DATE** – The first file saved after the specified date
- **No date** – The latest copy of the file

/DELETE_INTERVAL=*delta-time*

The **/DELETE_INTERVAL** qualifier specifies the duration of certain restore requests after their start of execution. The qualifier is applicable only to restores with a scheduling frequency

of **ONE_TIME_ONLY**. For these restores, the restore object (and any associated schedules and selections) are deleted after the delete interval has passed following the start of the restore execution. If you do not wish the restore to be deleted, specify **/KEEP**.

If not specified, the delete interval is 03-00:00:00 (3 days) for restores of frequency **ONE_TIME_ONLY**, and 0 for all other frequencies.

/DESCRIPTION="text"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/DESTINATION=disk:[dir]

The **/DESTINATION** qualifier specifies the target destination disk/directory location for the restore. Specify a disk and directory combination. If destination is not specified, the restore data is restored to the original disk location from where the data was saved.

The default is that no destination is specified, and the original save destination is used.

/ENVIRONMENT=environment-name

The **/ENVIRONMENT** qualifier specifies the environment object to use for the restore. Specify an environment name in the MDMS database.

If no environment is specified, one of the following defaults is used, in order of priority:

1. The first archive name suffixed by "_ENV"
2. "DEFAULT_ENV"

/EPILOGUE="string"

/NOEPILOGUE (Default)

The **/EPILOGUE** qualifier specifies a string to execute after each restore thread (i.e. each include list item specification in the restore). The string may be up to 80 characters and must be enclosed in quotes.

By default, no epilogue command is executed.

/EXCLUDE=(data-specification[,...])

/NOEXCLUDE (Default)

The **/EXCLUDE** qualifier is used to exclude files that otherwise would be included with the **/INCLUDE** qualifier. This is used to exclude files for save or restore when a more general wildcard specification is specified for the include. The exclude specification is stored in the default selection.

By default, no files are excluded.

/EXECUTION_NODES=(node-name[,...])

The **/EXECUTION_NODES** qualifier specifies one or more node names on which the restore request can be executed. Any one of the nodes can be the execution node for the request. The **/GROUPS** qualifier may also be specified.

If no nodes or groups are specified, the include specification uses the node on which the restore was created.

/EXPLICIT_INTERVAL="string"

/NOEXPLICIT_INTERVAL (Default)

When using scheduler type SCHEDULER, you can specify the scheduling frequency using **/EXPLICIT_INTERVAL**. Enter a string in a format known to the external scheduler. **/NOEXPLICIT_INTERVAL** clears the explicit interval. This qualifier is only applicable when **/FREQUENCY=EXPLICIT**.

By default, no explicit interval is defined.

/FREQUENCY=keyword

The **/FREQUENCY** qualifier specifies the scheduling frequency for this restore. Select one of the following options:

- **ONE_TIME_ONLY (Default)** – Scheduled one time only based on the base date or start date. This restore cannot be rescheduled, and the restore will automatically be deleted when the delete interval expires. This is the default.
- **ON_DEMAND** – Scheduled on demand when a start date (or base date) is entered. Can be rescheduled by entering a new start date (or base date).
- **DAILY** – Scheduled on a daily basis at the time contained within the base date. For example: 21-JAN-2001, 22-JAN-2001...
- **WEEKLY** – Scheduled on a weekly basis on anniversaries of the base date day (MON-FRI) and time. For example: 21-JAN-2001, 28-JAN-2001...
- **BIWEEKLY** – Scheduled on a biweekly (once every two weeks) basis on anniversaries of the base date day (MON-FRI) and time. For example: 21-JAN-2001, 04-FEB-2001...
- **SEMI_MONTHLY** – Scheduled twice a month based on the base date (1-31) and time, and either two weeks before or after the date. For example: 21-JAN-2001, 07-FEB-2001... +CHECK++
- **MONTHLY** – Scheduled monthly based on the base date (1-31) and time and subsequent monthly anniversaries. For example: 21-JAN-2001, 21-FEB-2001...
- **QUARTERLY** – Scheduled once every three months on the base date (1-31) and time, and three-month anniversaries thereafter. For example: 21-JAN-2001, 21-APR-2001...
- **SEMI_ANNUALLY** – Scheduled once every six months on the base date (1-31) and time, and six-month anniversaries thereafter. For example: 21-JAN-2001, 21-JUL-2001...
- **ANNUALLY** – Scheduled once every year on the base date (1-31) and time, and twelve-month anniversaries thereafter. For example: 21-JAN-2001, 21-JAN-2002...
- **NEVER** – Cannot be scheduled. This option can be used as a placeholder. The frequency must be changed to something else before the restore can execute.
- **EXPLICIT** – When using scheduler type SCHEDULER, this option utilizes the string in EXPLICIT_INTERVAL to determine the scheduling frequency.

- **CUSTOM** – When using scheduler types **INTERNAL** or **EXTERNAL**, this option utilizes a user-defined schedule object, which has the frequency defined internally. Use this option if none of the above scheduling frequencies is appropriate.

/GROUPS=(group-name[,...])
/NOGROUPS

Specifies the names of groups of execution nodes on which this restore can be run. Usually, only one group is specified. If neither **/EXECUTION_NODES** nor **/GROUPS** is specified, the node from which the command was issued is used as the node name.

/INCLUDE=(data-specification[,...])
/NOINCLUDE (Default)

The **/INCLUDE** qualifier allows you to enter a list of up to 24 data specifications to restore. The data specifications can be OpenVMS disks or files. However, all specifications must be of the same data select type, as specified in the **/DATA_SELECT_TYPE** qualifier. The include specification is stored in the default selection.

The include, exclude, and source node specifications, together with the data select type, are stored in a default selection object that has the name *restore-name_REST_SEL_DEF*.

This selection object can be modified independently to add such refinements as exclude lists, and additional selections can be added to any restore. **/NOINCLUDE** removes the include file specification.

By default, there is no include specification.

/INCREMENTAL
/NOINCREMENTAL (Default)

The **/INCREMENTAL** qualifier specifies that the restore should be an incremental restore. If not specified, the restore will be a full restore or selective restore based on the data-specification in the associated include specification or selections.

The default is **/NOINCREMENTAL**.

/KEEP

The **/KEEP** qualifier indicates that the restore is not to be automatically deleted, and clears the delete interval.

/OWNER_NAME=node::username
/OWNER_NAME=group::username
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/PROLOGUE=*"string"*

/NOPROLOGUE (Default)

The **/PROLOGUE** qualifier specifies a string to execute before each restore thread (i.e. each include list item specification in the restore). The string may be up to 80 characters and must be enclosed in quotes.

By default, no prologue command is executed.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/RESCHEDULE

The **/RESCHEDULE** qualifier is used with scheduler type SCHEDULER to force MDMS to reschedule the restore with the external scheduler. This is used when it is known that the scheduling job for this restore is no longer present in the scheduler and you wish to add it.

/SELECTIONS=(*selection*[,...])

/NOSELECTIONS

When using the default selection, this qualifier is not needed, as MDMS adds it automatically. However, if you want to add your own selections to this restore, you add them with the **/SELECTION** qualifier. By default a selection is added, but you can also remove selections with the **/REMOVE** qualifier, and replace selections with the **/REPLACE** qualifier. **/NOSELECTIONS** removes all selections from the restore.

/SEQUENCE_OPTION=*keyword*

The **/SEQUENCE_OPTION** qualifier defines the type of sequencing used during the restore operation. Select one of the following options:

- **SEQUENTIAL** – Performs each restore thread sequentially. Each restore thread must fully complete before the next one starts.
- **OVERLAPPED** – Starts the next restore thread as soon as the backup drive or device is available for use.

The default is **SEQUENTIAL**.

/SINCE_DATE=*date*

/NOSINCE_DATE

You can select files for restore based on the date archived after a certain date. If specified, the restore will apply to the first date archived after the specified date. You can specify date as an absolute date, as a combination of absolute and delta times, or as one of the following keywords:

TODAY (default), TOMORROW, or YESTERDAY.

/NOSINCE_DATE clears the since date and uses other date criteria to select files, as follows:

- BEFORE_DATE – The latest file saved before the specified date
- DATE_ARCHIVED – The latest file saved on the specified date
- No date – The latest copy of the file

/SKIP_TIME=delta-time

/NOSKIP_TIME (Default)

It is possible to run a restore at any time, including times when it is not normally scheduled. You may, for example, want to run a restore three hours early, or you may want to run an extra restore for some reason.

By default, if you run a restore at an unscheduled time, MDMS will also run it at its regularly scheduled time as well. If you do not want this, then enter a skip time when starting the restore at the non-scheduled time. The skip time will be added to the start time, and MDMS will not schedule the restore until at least after the skip_time value.

The default is **/NOSKIP_TIME**.

/START_DATE[=date]

/NOSTART_DATE

The **/START_DATE** qualifier is used to schedule a restore to start at an unscheduled time. The **/BASE_DATE** is used to specify the normal scheduled times. By default, the start date is set to the base date if not separately specified.

The date value for the **/START_DATE** is optional. If not specified, the default is now.

/NOSTART_DATE clears the next start date, and the restore will not be scheduled.

Examples

The following commands modifies a restore REST1, which uses archive REST1_ARC and environment REST1_ENV. The data to restore is NOD002::USER1\$: [SMITH...] *.*; *, which will be restored to the new destination USER2\$: [SMITH]. The data select type is VMS_FILES by default. The **/START** qualifier starts the restore immediately, and the restore is a ONE_TIME_ONLY:

```
$ MDMS SET RESTORE REST1 -
_$ /INCLUDE=(USER1$: [SMITH...] *.*; *) -
_$ /EXECUTION_NODES=NOD002 -
_$ /DESTINATION=USER2$: [SMITH] -
_$ /START/REPLACE
```

The following command modifies a restore REST2, the data selections of which contained in SELECT3 are removed from the selection list. This restore is now regularly scheduled on a semi-monthly basis on the first and fifteenth of each month at 1:00 am:

```
$ MDMS SET RESTORE REST2 -
_$ /SELECTIONS=SELECT3/REMOVE -
_$ /FREQUENCY=SEMI_MONTHLY -
_$ /BASE_DATE=01-JUL-2001:01:00:00
```


MDMS SET SAVE

MDMS SET SAVE — Modifies an existing save definition in the MDMS database. Saves were formerly known as save requests, archive requests, or archive transactions. Save is the new common name.

Format

```
MDMS SET SAVE [save-name] [, ...]
```

Equivalent ABS Command: **SET SAVE**

Parameters

save-name

Specifies the name of the save.

The maximum length of the save name is 39 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of save names may be entered.

Description

The **MDMS SET SAVE** command modifies an existing save definition in the MDMS database. Saves are used to copy online disk data to an archive for the purposes of data safety (backup), disaster recovery, or long-term, off-line archival.

The data specifications for saves can be supplied in one of two ways:

- Using the **/INCLUDE**, **/EXCLUDE**, **/DATA_SELECT_TYPE**, and **/SOURCE_NODE** qualifiers. These qualifiers can be used to specify a set of data specifications of a single data select type that can be saved. These are not positional qualifiers. This option is recommended for most saves, and this information is stored in a default selection that is mapped to the save by MDMS. The default selection name is *save-name_SAVE_SEL_DEF*.
- For more complex data specifications, you can create your own selections and map them to the save using the **/SELECTIONS** qualifier. This options gives you more granularity in data selection, and allows you to have mixed data select types in a single save.

Saves are normally scheduled to execute at regular intervals according to your site-specific needs. If you wish to use one of the supplied frequency options, MDMS creates a schedule and initializes it to appropriate values. The schedule for any save is called *save-name_SAVE_SCHED*.

Alternatively, if you wish to define a schedule not handled in the supplied frequencies, you can use one of the following options:

- For scheduler types **INTERNAL** and **EXTERNAL**, you can define **/FREQUENCY=CUSTOM**, and modify the schedule to your own specification.
- For scheduler type **SCHEDULER**, you can define **/FREQUENCY=EXPLICIT**, and define a string to pass the explicit schedule to the scheduler in the qualifier **/EXPLICIT_INTERVAL**.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

This command is restricted to nodes with an ABS license. The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The **/DELETE_INTERVAL** and **/KEEP** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))

/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/ARCHIVE=(*archive-name*[,*archive-name*])

The **/ARCHIVE** qualifier allows you to specify one or two archives associated with the save. The first archive name is applicable to full and selective threads in saves. If specified, the second archive name is associated with incremental threads in saves. If a second name is not specified, incremental

threads use the first archive name. For any thread, the catalog name from the first archive is always used. At least one archive name must be supplied for a save.

If an archive is not specified, the archive `SYSTEM_BACKUPS` is used by default.

***/BASE_DATE=*date**

/NOBASE_DATE

The base date is used as a starting point for saves that have a repeating schedule frequency. Subsequent schedules are applied by adding the multiples of the frequency to the base date. If the save is not to be repeated, then no base date is needed.

By default, if a base date is not specified but a start date is, the base date is set to the start date. If neither is specified, the base date is not set and the save is not scheduled for execution.

***/DATA_SELECT_TYPE=*keyword**

The ***/DATA_SELECT_TYPE*** qualifier specifies the type of data to save in the default selection. Select one of the following options:

- `VMS_FILES`
- `VMS_SAVESET`

If not specified, MDMS attempts to determine the data select type based on the include file syntax. If MDMS cannot determine this, `VMS_FILES` is used. The data select type specification is stored in the default selection.

***/DELETE_INTERVAL=*delta-time**

The ***/DELETE_INTERVAL*** qualifier specifies the duration of certain saves after their start of execution. The qualifier is applicable only to saves with a scheduling frequency of `ONE_TIME_ONLY`. For these saves, the save object (and any associated schedules and selections) are deleted after the delete interval has passed following the start of the save execution. If you do not wish the save to be deleted, specify ***/KEEP***.

If not specified, the delete interval is 03-00:00:00 (3 days) for saves of frequency `ONE_TIME_ONLY`, and 0 for all other frequencies.

/DESCRIPTION="*text*"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify `""`.

***/ENVIRONMENT=*environment-name**

The ***/ENVIRONMENT*** qualifier specifies the environment object to use for the save. Specify an environment name in the MDMS database.

If no environment is specified, one of the following defaults is used, in order of priority:

1. The first archive name suffixed by `"_ENV"`
2. `"DEFAULT_ENV"`

/EPILOGUE="string"
/NOEPILOGUE (Default)

The **/EPILOGUE** qualifier specifies a string to execute after each save thread (i.e. each include list item specification in the save). The string may be up to 80 characters and must be enclosed in quotes.

By default, no epilogue command is executed.

/EXCLUDE=(data-specification[,...])
/NOEXCLUDE (Default)

The **/EXCLUDE** qualifier is used to exclude files that otherwise would be included with the **/INCLUDE** qualifier. This is used to exclude files for save or restore when a more general wildcard specification is specified for the include. The exclude specification is stored in the default selection.

By default, no files are excluded.

/EXECUTION_NODES=(node-name[,...])

The **/EXECUTION_NODES** qualifier specifies one or more node names on which the save request can be executed. Any one of the nodes can be the execution node for the request. The **/GROUPS** qualifier may also be specified.

If no nodes or groups are specified, the include specification uses the node on which the save was created.

/EXPLICIT_INTERVAL="string"
/NOEXPLICIT_INTERVAL (Default)

When using scheduler type SCHEDULER, you can specify the scheduling frequency using **/EXPLICIT_INTERVAL**. Enter a string in a format known to the external scheduler. **/NOEXPLICIT_INTERVAL** clears the explicit interval. This qualifier is only applicable when **/FREQUENCY=EXPLICIT**.

By default, no explicit interval is defined.

/FREQUENCY=keyword

The **/FREQUENCY** qualifier specifies the scheduling frequency for this save. Select one of the following options:

- **ONE_TIME_ONLY (Default)** – Scheduled one time only based on the base date or start date. This save cannot be rescheduled, and the save will automatically be deleted when the delete interval expires. This is the default.
- **ON_DEMAND** – Scheduled on demand when a start date (or base date) is entered. Can be rescheduled by entering a new start date (or base date).
- **DAILY** – Scheduled on a daily basis at the time contained within the base date. For example: 21-JAN- 2001, 22-JAN-2001...
- **DAILY_FULL_WEEKLY** – Scheduled on a daily basis, with a full save on the base date and weekly anniversaries, and incremental saves on other days. For example: 21-JAN-2001, 22-JAN-2001...

- WEEKLY – Scheduled on a weekly basis on anniversaries of the base date day (MON-FRI) and time. For example: 21-JAN-2001, 28-JAN-2001...
- BIWEEKLY – Scheduled on a biweekly (once every two weeks) basis on anniversaries of the base date day (MON-FRI) and time. For example: 21-JAN-2001, 04-FEB-2001...
- SEMI_MONTHLY – Scheduled twice a month based on the base date (1-31) and time, and either two weeks before or after the date. For example: 21-JAN-2001, 07-FEB-2001... +CHECK++
- MONTHLY – Scheduled monthly based on the base date (1-31) and time and subsequent monthly anniversaries. For example: 21-JAN-2001, 21-FEB-2001...
- QUARTERLY – Scheduled once every three months on the base date (1-31) and time, and three-month anniversaries thereafter. For example: 21-JAN-2001, 21-APR-2001...
- SEMI_ANNUALLY – Scheduled once every six months on the base date (1-31) and time, and six-month anniversaries thereafter. For example: 21-JAN-2001, 21-JUL-2001...
- ANNUALLY – Scheduled once every year on the base date (1-31) and time, and twelve-month anniversaries thereafter. For example: 21-JAN-2001, 21-JAN-2002...
- LOG_2 – Scheduled daily based on a full and incremental algorithm to optimize save/restore times. For example: 21-JAN-2001, 22-JAN-2001... (see the [VSI Archive Backup System for OpenVMS User Guide](https://docs.vmssoftware.com/vsi-abs-user-guide/) [https://docs.vmssoftware.com/vsi-abs-user-guide/])
- LOG_3 – Scheduled daily based on a full and incremental algorithm to optimize save/restore times. For example: 21-JAN-2001, 22-JAN-2001... (see the [VSI Archive Backup System for OpenVMS User Guide](https://docs.vmssoftware.com/vsi-abs-user-guide/) [https://docs.vmssoftware.com/vsi-abs-user-guide/])
- NEVER – Cannot be scheduled. This option can be used as a placeholder. The frequency must be changed to something else before the save can execute.
- EXPLICIT – When using scheduler type SCHEDULER, this option utilizes the string in EXPLICIT_INTERVAL to determine the scheduling frequency.
- CUSTOM – When using scheduler types INTERNAL or EXTERNAL, this option utilizes a user-defined schedule object, which has the frequency defined internally. Use this option if none of the above scheduling frequencies is appropriate.

/GROUPS=(group-name[,...])

/NOGROUPS

Specifies the names of groups of execution nodes on which this save can be run. Usually, only one group is specified. If neither **/EXECUTION_NODES** nor **/GROUPS** is specified, the node from which the command was issued is used as the node name.

/INCLUDE=(data-specification[,...])

/NOINCLUDE (Default)

The **/INCLUDE** qualifier allows you to enter a list of up to 24 data-specifications to restore. The data specifications can be OpenVMS disks or files. However, all specifications must be of the same data select type, as specified in the **/DATA_SELECT_TYPE** qualifier.

The include, exclude, and source node specifications, together with the data select type, are stored in a default selection object that has the name *save-name_SAVE_SEL_DEF*.

This selection object can be modified independently to add such refinements as exclude lists, and additional selections can be added to any save. **/NOINCLUDE** removes the include file specification.

By default, there is no include specification.

/INCREMENTAL

/NOINCREMENTAL (Default)

The **/INCREMENTAL** qualifier specifies that the save should be an incremental save. If not specified, the save will be a full save or selective save based on the data-specification in the associated include specification or selections.

The default is **/NOINCREMENTAL**.

/KEEP

The **/KEEP** qualifier indicates that the save is not to be automatically deleted, and clears the delete interval.

/OWNER_NAME=node::username

/OWNER_NAME=group::username

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/PROLOGUE="string"

/NOPROLOGUE (Default)

The **/PROLOGUE** qualifier specifies a string to execute before each save thread (i.e. each include list item specification in the save). The string may be up to 80 characters and must be enclosed in quotes.

By default, no prologue command is executed.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/RESCHEDULE

The **/RESCHEDULE** qualifier is used with scheduler type **SCHEDULER** to force MDMS to reschedule the save with the external scheduler. This is used when it is known that the scheduling job for this save is no longer present in the scheduler and you wish to add it.

/SELECTIONS=(*selection*[,...])**/NOSELECTIONS**

When using the default selection, this qualifier is not needed, as MDMS adds it automatically. However, if you want to add your own selections to this save, you add them with the **/SELECTION** qualifier. By default a selection is added, but you can also remove selections with the **/REMOVE** qualifier, and replace selections with the **/REPLACE** qualifier. **/NOSELECTIONS** removes all selections from the save.

/SEQUENCE_OPTION=*keyword*

The **/SEQUENCE_OPTION** qualifier defines the type of sequencing used during the save operation. Select one of the following options:

- **SEQUENTIAL** – Performs each save thread sequentially. Each save thread must fully complete before the next one starts.
- **OVERLAPPED** – Starts the next save thread as soon as the backup drive or device is available for use.

The default is **SEQUENTIAL**.

/SKIP_TIME=*delta-time***/NOSKIP_TIME (Default)**

It is possible to run a save at any time, including times when it is not normally scheduled. You may, for example, want to run a save three hours early, or you may want to run an extra save for some reason.

By default, if you run a save at an unscheduled time, MDMS will also run it at its regularly scheduled time as well. If you do not want this, then enter a skip time when starting the save at the non-scheduled time. The skip time will be added to the start time, and MDMS will not schedule the save until at least after the *delta-time* value.

The default is **/NOSKIP_TIME**.

/START_DATE[=*date*]**/NOSTART_DATE**

The **/START_DATE** qualifier is used to schedule a save to start at an unscheduled time. The **/BASE_DATE** is used to specify the normal scheduled times. By default, the start date is set to the base date if not separately specified.

The date value for the **/START_DATE** is optional. If not specified, the default is now. **/NOSTART_DATE** clears the next start date, and the save will not be scheduled.

For example, when the following command is issued, the start time is set to the system time on the current MDMS database server node, irrespective of whether the command is issued from server or client:

```
$ MDMS SET SAVE /START="F$TIME"
```

When the following command is issued, the start time is set to the system time on the node on which this command is issued:

```
$ MDMS SET SAVE /START="TODAY"
```

Examples

The following command modifies save SAVE1 to change the base date to 22-OCT-2001:01:00:00 and ask to reschedule the save. A prologue and epilogue are specified, and a new selection object is added:

```
$ MDMS SET SAVE SAVE1 -  
_ $ /BASE_DATE=22-OCT-2001:01:00:00/RESCHEDULE -  
_ $ /PROLOGUE="@USER1:[ABS] SAVE1_PROLOGUE" -  
_ $ /EPILOGUE="@USER1:[ABS] SAVE1_EPILOGUE" -  
_ $ /SELECTIONS=SELECT4/ADD
```

The following command modifies save SAVE2 to add an access control for NOD001::SMITH to set and show this save. It also requests the save for immediate execution:

```
$ MDMS SET SAVE SAVE2 -  
_ $ /ACCESS_CONTROL=(USER=NOD001::SMITH, ALLOW=(SET,SHOW)) -  
_ $ /START
```

MDMS SET SCHEDULE

MDMS SET SCHEDULE — Modifies an existing schedule definition in the MDMS database. Schedule is a new object type.

Format

```
MDMS SET SCHEDULE [schedule-name] [, ...]
```

Equivalent ABS Command: None.

Parameters

schedule-name

Specifies the name of the schedule.

The maximum length of the schedule name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of schedule names may be entered.

Description

The **MDMS SET SCHEDULE** command modifies an existing schedule definition in the MDMS database. You can use schedules to execute any DCL command (including MDMS commands) at regular intervals.

Schedules are used by MDMS to execute saves and restores. You can only modify CUSTOM schedules using this command.

You can also modify any schedules that you specifically created.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/AFTER_SCHEDULE=(NAME=*schedule-name*, WHEN=*option*)
/NOAFTER_SCHEDULE

The **/AFTER_SCHEDULE** qualifier specifies that this schedule is scheduled for execution after the AFTER_SCHEDULE name, under the circumstances defined in WHEN. This can be used to serialize a series of operations one after the other, and conditionally execute this schedule depending on the status of the AFTER_SCHEDULE.

The SCHEDULE should be a schedule name defined in the MDMS database. For WHEN, select one of the following:

ALL (Default)	Unconditionally execute this schedule when the AFTER_SCHEDULE completes
ERROR	Execute this schedule if the AFTER_SCHEDULE completed with ERROR or FATAL status
FATAL	Execute this schedule if the AFTER_SCHEDULE completed with FATAL status
NONE	Do not execute this schedule (can be used as a placeholder)
SUCCESS	Execute this schedule if the AFTER_SCHEDULE completed with SUCCESS or INFORMATIONAL status
WARNING	Execute this schedule if the AFTER_SCHEDULE completed with WARNING, ERROR or FATAL status

If an **/AFTER_SCHEDULE** name is specified, the default WHEN option is ALL.

/NOAFTER_SCHEDULE removes the dependency to execute after another schedule.

/COMMAND="*string*"

The **/COMMAND** qualifier specifies the command to submit to DCL when the schedule executes. Specify a valid DCL command line in quotes. This qualifier must be entered for the schedule to do any useful work.

/DATES=(*date*[,...])
/NODATES (Default)

The **/DATES** qualifier specifies on which days of the month you wish the schedule to execute. Valid values are 1-31. You can enter a date, a list of dates, a range of dates, and a list of ranges (e.g. 1-7, 15-21). **/NODATES** indicates that all dates are valid (1-31) and other criteria are for scheduling. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the dates list.

The default is **/NODATES**.

/DAYS=(*day*[,...])
/NODAYS (Default)

The **/DAYS** qualifier specifies on which days of the week you wish the schedule to execute. Valid values use at least the first three letters of the days of the week (English only). You can enter a day, a list of days, a range of days, and a list of ranges (e.g. MON-WED, SAT-SUN). **/NODAYS** indicates that all day are valid (MON-SUN) and other criteria are used for scheduling. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the days list.

The default is **/NODAYS**.

/DESCRIPTION="text"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/EXCLUDE=(date[,...])

/NOEXCLUDE (Default)

The **/EXCLUDE** qualifier specifies specific dates that you wish to exclude from execution, even though the other scheduling parameters would normally schedule on that date. This can be used to exclude scheduled operations on holidays for example. Enter one or more dates in OpenVMS date format (04-JUL-2001), separated by commas. You can specify dates up to 9 years in the future. **/NOEXCLUDE** removes all dates from the exclude list. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the exclude list.

By default, no exclude dates are defined.

/INCLUDE=(date[,...])

/NOINCLUDE (Default)

The **/INCLUDE** qualifier specifies specific dates that you wish to include for execution, even though the other scheduling parameters would normally not schedule on that date. This can be used to include operations on special days that require special attention. Enter one or more dates in OpenVMS date format (31-DEC-2001), separated by commas. You can specify dates up to 9 years in the future. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the include list.

By default, no include dates are defined.

/MONTHS=(month[,...])

/NOMONTHS (Default)

The **/MONTHS** qualifier specifies on which months of the year you wish the schedule to execute. Valid values use at least the first three letters of the months of the year (English only). You can enter a month, a list of months, a range of months, and a list of ranges (e.g. JAN-MAR, JULSEP). **/NOMONTHS** indicates that all months are valid (JAN-DEC) and other criteria are used for scheduling. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the months list.

The default is **/NOMONTHS**.

/OWNER_NAME=node::username

/OWNER_NAME=group::username

/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If

you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/TIMES=(time[,...])

/NOTIMES (Default)

The **/TIMES** qualifier specifies the times of the day you wish the schedule to execute. Enter from 1 to 100 times in OpenVMS format, separated by commas (e.g. 08:00, 10:00, 12:00, 14:00, 16:00). **/NOTIMES** clears out all times and effectively disables the schedule from execution. **/TIMES=00:00** signifies midnight.

By default, no times are set.

/WEEKS=(week[,...])

/NOWEEKS (Default)

The **/WEEKS** qualifier specifies on which weeks of the year you wish the schedule to execute. Valid values are 1-53. You can enter a week, a list of weeks, a range of weeks and a list of ranges (e.g. 1-7, 15-21, 31-53). **/NOWEEKS** indicates that all weeks are valid (1-53) and other criteria are used for scheduling. You can use the **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers to manipulate the weeks list.

The default is **/NOWEEKS**.

Examples

The following command modifies schedule SCHED1 to run every Saturday at 21:00 by issuing the **@USER2: [SMITH] SCHED1.COM** command procedure:

```
$ MDMS SET SCHEDULE SCHED1 -
_$ /DAYS=SAT -
_$ /TIMES=21:00 -
_$ /COMMAND="@USER2: [SMITH] SCHED1.COM"
```

The following command modifies schedule SCHED2 to execute every month on the first Sunday of the month at noon by issuing an **MDMS DELETE SAVE** command:

```
$ MDMS SET SCHEDULE SCHED2 -
_$ /NOMONTHS -
_$ /DAYS=SUN -
_$ /DATES=(1-7) -
_$ /TIMES=12:00 -
```

```
_$_ /COMMAND="MDMS DELETE SAVE *MONTH*"
```

MDMS SET SELECTION

MDMS SET SELECTION — Modifies an existing selection definition in the MDMS database. Selection is a new object type.

Format

```
MDMS SET SELECTION [selection-name] [, ...]
```

Equivalent ABS Command: None.

Parameters

selection-name

Specifies the name of the selection.

The maximum length of the selection name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of selection names may be entered.

Description

The **MDMS SET SELECTION** command modifies an existing selection definition in the MDMS database. Selections are used to specify the data to be saved or restored in those objects.

Under most circumstances, you can specify the data to be saved or restored in the appropriate save or restore object, and MDMS will implicitly create a default selection. However, there may be times that you wish to create your own selections and add them to the relevant save or restore object. This may be because you wish to specify a mixture of data select types in a save/restore or specific file dates in a save.

You may modify both user-created and default selections using this command.

Many of the qualifiers for selections map to the positional qualifiers on saves and restores with the pre-V4 ABS **SAVE** and **RESTORE** commands.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/AGENT_QUALIFIERS="*string*"
/NOAGENT_QUALIFIERS (Default)

The **/AGENT_QUALIFIERS** qualifier allows you to specify a backup-agent specific qualifier or set of qualifiers. This qualifier is valid for all data select types. Specify up to 80 characters enclosed in quotes. **/NOAGENT_QUALIFIERS** removes all the agent qualifiers.

By default, no agent qualifiers are defined.

/BEFORE_DATE=*date*
/NOBEFORE_DATE (Default)

The **/BEFORE_DATE** is used to select files that have a date before the specified date. The type of date in question is specified by the **/DATE_TYPE** qualifier. If **/NOBEFORE_DATE** is specified, files are not selected using a before date.

/BEFORE_DATE is applicable only to saves.

The default is that no before date is specified.

/CONFLICT_OPTIONS=option

The **/CONFLICT_OPTIONS** is used to determine the disposition of a file if an existing file already exists. This qualifier is applicable only to restores. Select one of the following:

- **NEW_VERSION** – Create a new version of the file
- **OVERLAY_VERSION** – Overlay the data in the current version of the file, but maintain the existing file header
- **REPLACE_VERSION** – Delete the existing version and replace it with the archive version
- **RETAIN_VERSION** (Default) – Leave the existing version and do not restore the version from the archive

If not specified, the default is **RETAIN_VERSION**.

/DATA_SELECT_TYPE=keyword

The **/DATA_SELECT_TYPE** qualifier specifies the type of data to restore in the default selection. Select one of the following options:

- **VMS_FILES**
- **VMS_SAVESET**

If not specified, MDMS attempts to determine the data select type based on the include file syntax. If MDMS cannot determine this, **VMS_FILES** is used.

/DATE_TYPE

The **/DATE_TYPE** qualifier specifies which date in a file is used using **/BEFORE_DATE** and/or **/SINCE_DATE** in a selection. Select one of the following date types:

- **BACKUP** – The BACKUP date is used for selection
- **EXPIRED** – The expiration date is used for selection. This can also be configured as last access date
- **CREATED** – The creation date is used for selection.
- **MODIFIED** (Default) – The revision date is used for selection

If not specified, **MODIFIED** is the default.

/DESCRIPTION="text"

Comments about the object. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. To clear the description, specify "".

/EXCLUDE=(data-specification[,...])
***/NOEXCLUDE* (Default)**

The **/EXCLUDE** qualifier is used to exclude files that otherwise would be included with the **/INCLUDE** qualifier. This is used to exclude files for save or restore when a more general wildcard specification is specified for the include.

By default, no files are excluded.

/INCLUDE=(*data-specification*[,...])
/NOINCLUDE

The **/INCLUDE** qualifier allows you to enter a list of up to 24 data-specifications to restore. The data specifications can be OpenVMS disks or files. However, all specifications must be of the same data select type, as specified in the **/DATA_SELECT_TYPE** qualifier.

/NOINCLUDE clears the include specification.

See [MDMS CREATE RESTORE](#) for the note on maintaining file directory structure during the Restore operation.

/OWNER_NAME=*node::username*
/OWNER_NAME=*group::username*
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/SINCE_DATE[=*date*]
/NOSINCE_DATE (Default)

The **/SINCE_DATE** is used to select files that have a date after the specified date. The type of date in question is specified by the **/DATE_TYPE** qualifier. If **/NOSINCE_DATE** is specified, files are not selected using a since date.

/SINCE_DATE is only applicable to saves.

The default is that no since date is specified.

Examples

The following command modifies selection **SEL1** to include files in **USER1 : [BACON] *.* ; *** that have been modified since 24-MAY-2025. It excludes all **.LOG** and **.TMP** files in that directory:


```
$ MDMS SET SELECTION SEL1 -  
_ $ /INCLUDE=(USER1:[BACON]*.*;*) -  
_ $ /EXCLUDE=(USER1:[BACON]*.LOG;* , *.TMP;*) -  
_ $ /SINCE=24-MAY-2025
```

The following command modifies selection SEL2 to include files in /usr/smith/ on node XTRIX:

```
$ MDMS SET SELECTION SEL2 -  
_ $ /INCLUDE=(/usr/smith/) -  
_ $ /SOURCE_NODES=XTRIX
```

MDMS SET SERVER

MDMS SET SERVER — Resets server conditions. Currently, it only supports resetting the MDMS logging file.

Format

```
MDMS SET SERVER /RESET_LOG
```

Equivalent STORAGE Command: None.

Parameters

None.

Description

The **MDMS SET SERVER** command resets server conditions. Currently it only supports resetting the MDMS logging file.

You can specify to which node the reset will apply, or all nodes in the domain, using qualifiers. If you do not specify any of the node-related qualifiers, the log file on the issuing client node is reset.

Privileges

The request requires MDMS_SET_ALL.

Restrictions

The **/ALL**, **/DBSERVER**, and **/NODE** qualifiers are mutually exclusive. If none are specified, the log on the issuing (client) node is reset.

Qualifiers

/ALL

This qualifier specifies that you want to reset the log file on all nodes in the domain.

/DBSERVER

This qualifier specifies that you want to reset the database server's log file. This is the log file that contains the most complete information.

/KEEP=number

This qualifier specifies that you wish to purge the associated log files and keep the specified number of versions. If you omit the **/KEEP** qualifier, then the logs are not purged.

/NODE=node-name

This qualifier specifies that you want to reset the log file on the specified node.

/RESET_LOG

This qualifier closes the current log file and opens a new one. This qualifier is required.

Examples

The following command closes the current log file and opens a new version on the database server node. It also purges the MDMS log files on the database server node, so that 5 versions remain after the reset:

```
$ MDMS SET SERVER /RESET_LOG/DBSERVER/KEEP=5
```

The new files will be opened at the location designated by the logical name MDMS\$LOGFILE_LOCATION on the affected nodes.

MDMS SET VOLUME

MDMS SET VOLUME — Modifies a volume definition in the MDMS volume database.

Format

```
MDMS SET VOLUME [volume-id] [, ...]
```

Equivalent STORAGE Command: **STORAGE SET VOLUME**

Parameters

volume-id

Specifies the volume ID of the volume to be added. The volume ID is the external label for the volume. Alternatively, a volume range (separated by a dash) may be specified. A volume range is a numeric range for up to the last *n* characters of the volume ID. While specifying the volume range, make sure that:

- There are the same number of characters on both sides of the dash.
- The data type of the corresponding position on either side of the dash must be the same.

Example ranges are (A00001-A99999) and (AB0001-AB9999).

The maximum length of the volume ID is 6 characters. Valid characters are " , ! , " , % - ? , A - Z , _ " .

A comma-separated list of volume IDs or ranges may be entered.

Description

The **MDMS SET VOLUME** command modifies a volume definition in the MDMS volume database.

Privileges

The request requires MDMS_SET_ALL, MDMS_SET_VOLUME, MDMS_SET_POOL, or MDMS_SET_OWN.

If the user is modifying a volume that is allocated to them, MDMS_SET_OWN is sufficient. If the user is modifying a volume that belongs to a pool to which they are authorized, MDMS_SET_POOL is required. For any other volume, MDMS_SET_VOLUME or MDMS_SET_ALL is required.

Restrictions

The **/ADD**, **/REMOVE**, and **/REPLACE** qualifiers are mutually exclusive. If none are specified, attributes are added to list attributes by default.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ACCESS_CONTROL=(USERNAME=*user-id*, ALLOW=(*option*[,...]))
/NOACCESS_CONTROL (Default)

Access control allows the system administrator to restrict access to the object to all or specific users on specific nodes or groups. Select one or more of the following options to allow the operation:

- SET – Allows setting the attributes of the object
- SHOW – Allows showing the object
- DELETE – Allows deleting the object
- READ – Allows restoring data using the object
- WRITE – Allows saving data using the object
- EXECUTE – Allows execution operations using the object
- CONTROL – Allows setting the access control on the object

By default, access controls are added to the list of access controls; only one user can be added per command. However, you can remove users using the **/REMOVE** qualifier and replace the entire access control using the **/REPLACE** qualifier. You can remove all access controls on the object by specifying **/NOACCESS_CONTROL**.

The *user-id* should be in the format *node::username* or *group::username*. Wildcards are supported in the node, group, and username portions.

Access control restrictions are in addition to the normal MDMS rights allowing operations on an object. If no access control is defined, then the default domain access control is applied.

/ACCOUNT="*text*"

Defines the account name of the volume. The account name may be up to 31 characters. If it contains spaces, it must be enclosed in quotation marks. Specify "" to clear the account name.

This qualifier requires the right MDMS_SET_PROTECTED.

/ADD (Default)

The **/ADD** qualifier works in conjunction with certain qualifiers that accept lists and adds the specified attributes to the existing list. This is the default when specifying list items.

/ALLOCATED_DATE=*date*
/NOALLOCATED_DATE

Specifies the date the volume was allocated. Normally this is set by MDMS. Specify a VMS absolute date and time. Specify **/NOALLOCATED_DATE** to clear the allocated date.

This qualifier requires the right MDMS_SET_PROTECTED.

/AVAILABLE

The **/AVAILABLE** qualifier moves a volume from the UNAVAILABLE state to the state it was previously in prior to the UNAVAILABLE state. The volume may then be moved into the TRANSITION or FREE state if the scratch date and/or transition time have expired.

/BLOCK_FACTOR=*number*

Specifies the block factor for the volume. The default is a block factor of zero.

/BRAND="*text*"

The media manufacturer. The maximum length of the brand name is 31 characters. If it contains spaces it must be contained in quotation marks. Specify "" to clear the brand.

/CLEANED_DATE=*date*
/NOCLEANED_DATE

This qualifier specifies the date the volume was last cleaned and is entered as a VMS absolute time. Specify **/NOCLEANED_DATE** to clear the cleaned date.

/CREATION_DATE=*date*
/NOCREATION_DATE

The date the volume is created. This attribute is set by MDMS, but may be overridden if necessary.

This qualifier requires the right MDMS_SET_PROTECTED.

/DEALLOCATED_DATE=*date*
/NODEALLOCATED_DATE

This qualifier specifies the actual deallocation date for the volume. Specify a VMS absolute time. This date is normally set by MDMS. This qualifier requires the right MDMS_SET_PROTECTED.

/DESCRIPTION="*text*"

Comments about the volume. If the text contains spaces, then it must be enclosed within quotation marks. The length of the description can range from 0 to 255 characters. Specify "" to clear the description.

/DRIVE=*drive-name*
/NODRIVE

This is a protected field that should be modified only to recover on error. Use the **LOAD** and **UNLOAD** commands to manipulate the drive name under normal operation. This qualifier specifies

the drive that the volume currently resides in or last resided in. This is normally set up by MDMS. To clear the drive, specify **/NODRIVE**.

This qualifier requires the right MDMS_SET_PROTECTED.

/FORMAT=*keyword*

Specifies the format of the tape. Possible values are as follows:

ASCII
BACKUP
EBCDIC
NONE (Default)
RMUBACKUP

/FREED_DATE=*date*

/NOFREED_DATE

Specifies the date the volume was last freed (i.e. put in the FREE state). Specify an OpenVMS absolute date and time. This is normally set up by MDMS. To clear the freed date, specify **/NOFREED_DATE**.

This qualifier requires the right MDMS_SET_PROTECTED.

/INITIALIZED_DATE=*date*

/NOINITIALIZED_DATE

Specifies the date the volume was last initialized. Specify a VMS absolute date and time. This is normally set up by MDMS. To clear the initialized date, specify **/NOINITIALIZED_DATE**.

This qualifier requires the right MDMS_SET_PROTECTED.

/IO_ERROR_COUNT=*number*

This qualifier allows you to set the number of I/O errors on the volume. The default value is zero.

/JOB_NAME="*text*"

This qualifier allows you to specify the last job that accessed the volume. The job name can be from 0 to 31 characters. If it contains spaces, it must be enclosed in quotation marks. Specify "" to clear the job name.

This qualifier requires the right MDMS_SET_PROTECTED.

/JUKEBOX=*jukebox-name*

/NOJUKEBOX

This is a protected field that should be modified only to recover on error. Use the **MOVE VOLUME** command to manipulate the jukebox name under normal operation. This qualifier allows you to specify that the volume is currently residing or last resided in the specified jukebox. The maximum length of a jukebox name is 31 characters. This is normally set up by MDMS. To clear the jukebox name, specify **/NOJUKEBOX**.

This qualifier requires the right MDMS_SET_PROTECTED.

/LAST_ACCESS_DATE=*date*
/NOLAST_ACCESS_DATE

Specifies the date the volume was last accessed. Specify a VMS absolute date and time. This is normally set up by MDMS. To clear the last access date, specify **/NOLAST_ACCESS_DATE**.

This qualifier requires the right MDMS_SET_PROTECTED.

/MAGAZINE=*magazine-name*
/NOMAGAZINE

This is a protected field that should be modified only to recover on error. Use the **MOVE VOLUME** command to manipulate the magazine name under normal operation. This qualifier specifies the magazine name if the volume resides in a magazine. This is normally set up by MDMS. To clear the magazine name, specify **/NOMAGAZINE**.

This qualifier requires the right MDMS_SET_PROTECTED.

/MEDIA_TYPES=(*media-type*[,...])
/NOMEDIA_TYPES

The media type qualifier allows you to add the media types that the volume can support. Multiple media types are supported prior to the volume being initialized. After initialization, a volume can only support one media type. The **/REMOVE** or **/REPLACE** qualifier can be used to remove or replace objects in the list, rather than adding them by default. To specify the volume supports no media types, enter **/NOMEDIA_TYPES**. If a volume is created with no media types, the default media type from the domain record is used.

/MOUNT_COUNT=*number*

Specifies the number of times the volume has been loaded by MDMS. Normally set up by MDMS. The default mount count is zero.

/OFFSITE=([LOCATION=*location*]**,**[NO]DATE[=*date*]**)**
/NOOFFSITE

This qualifier specifies the date that the volume is to be taken off site and the off-site location. The location field is required when using the **/OFFSITE** qualifier if no location has been previously specified. The LOCATION keyword cannot be negated and if specified must point to a valid location object. Specify a VMS absolute or delta time in the date field. The NODATE keyword may be used to remove the date. This has the effect of disabling the schedule for the volume, while retaining the location. To clear the off-site date, specify **/NOOFFSITE**. If a volume is under magazine control, the **/OFFSITE** qualifier is not allowed. The volume uses the values in the magazine. If the **/MAGAZINE** qualifier is used on the **SET VOLUME** command when an existing **/OFFSITE** location and date are set, the values are cleared.

The default off-site location is that stored in the domain object.

/ONSITE=([LOCATION=*location*]**,**[NO]DATE[=*date*]**)**
/NOONSITE

This qualifier specifies the date that the volume is to be brought back on site and the on-site location. The location field is required when using the **/ONSITE** qualifier if no location has been previously specified. The LOCATION keyword cannot be negated and if specified must point to a valid location object. Specify a VMS absolute or delta time in the date field. The NODATE keyword may be used to remove the date. This has the effect of disabling the schedule for the volume, while retaining the

location. To clear the on-site date specify **/NOONSITE**. If a volume is under magazine control, the **/ONSITE** qualifier is not allowed. The volume uses the values in the magazine. If the **/MAGAZINE** qualifier is used on the **SET VOLUME** command when an existing **/ONSITE** location and date are set, the values are cleared.

The default on-site location is that stored in the domain object.

/OWNER_NAME=node::username
/OWNER_NAME=group::username
/NOOWNER_NAME

The **/OWNER_NAME** qualifier specifies the owner of the object. Specify an owner as *node::username* or *group::username*.

The *node*, *group*, and *username* portions may be wildcarded.

The owner of an object has default access control of all allowances to the object, regardless of whether a specific access control entry is present or the state of the default domain access control. If you wish to remove this unconditional owner access, specify **/NOOWNER_NAME**. Note that being an owner of an object overrides only the access control; the appropriate MDMS rights for a request on the object must still be honored.

By default, the owner of an object is the *group-name::username* of the user entering the **CREATE** command on the object from a node in the cluster.

/OWNER=[owner-name]
/NOOWNER

This qualifier specifies the owner UIC of a volume. The owner UIC field must be a UIC in the format [USER] or [group, user]. This is normally set up by MDMS on allocate volume. To clear the owner UIC field, specify **/NOOWNER**.

This qualifier requires the right MDMS_SET_PROTECTED.

/PLACEMENT=keyword

This is a protected field that should be modified only to recover on error. Use the **MOVE**, **LOAD**, or **UNLOAD** commands to manipulate the placement field under normal operation. This qualifier defines the current placement of the volume. This is normally managed by MDMS. Select one of the following options:

DRIVE
ONSITE
MAGAZINE
JUKEBOX
OFFSITE
MOVING

If a magazine name is specified on the **/MAGAZINE** qualifier, the volume placement can be in one of three states: MAGAZINE, DRIVE, or MOVING.

During a **MOVE**, **LOAD**, or **UNLOAD** operation, a volume's placement may be set to MOVING, indicating that the volume is being moved. If a volume is in a magazine, it is set to MOVING when the volume is being loaded or unloaded to/from a drive.

This qualifier requires the right MDMS_SET_PROTECTED.

/POOL=pool-name
/NOPPOOL (Default)

The pool in which the volume belongs. The maximum length of the pool name is 31 characters. Spaces are not allowed in the pool name.

/PREINITIALIZED
/NOPREINITIALIZED (Default)

This qualifier specifies whether the volume has been initialized before creation. If **/PREINITIALIZED** is specified, the volume is placed in the FREE state rather than the UNINITIALIZED state.

/PROTECTION=protection

The protection code for the volume. Use the standard OpenVMS protection code format.

/PURCHASED_DATE=date
/NOPURCHASED_DATE

The date when the volume was purchased. The date should be specified as an OpenVMS absolute time. Specify **/NOPURCHASED_DATE** to clear the purchased date.

/RECLength=number

This qualifier specifies the record length used on the volume. The default record length is zero.

/RELEASE

This qualifier puts the volume into the FREE state from the TRANSITION state. It is not valid to release an allocated volume. This qualifier always operates on volume sets and works only if all the volumes in the volume set are in TRANSITION state.

/REMOVE

The **/REMOVE** qualifier works in conjunction with certain qualifiers that accept lists and removes specified attributes from them.

/REPLACE

The **/REPLACE** qualifier works in conjunction with certain qualifiers that accept lists and replaces the existing attributes list with the specified list. By default, attributes are added to the existing list.

/RETAIN

This qualifier puts the volume that is in the FREE state or TRANSITION state back into the allocated state with the former owner as the current owner. This qualifier always operates on volume sets.

/SCRATCH_DATE=date
/NOSCRATCH_DATE

The **/SCRATCH_DATE** qualifier specifies the planned date to return the volume from the ALLOCATED state to either the TRANSITION state or the FREE state. Specify **/NOSCRATCH_DATE** if the volume should not automatically transition from the ALLOCATED state.

/SLOTS=(*range*[,...])
/NOSLOTS

This qualifier specifies the jukebox or magazine slot that the volume currently resides in. Specify a number in the range of the jukebox or magazine. This is normally set up by MDMS. If the volume does not currently reside in a jukebox or magazine slot, specify **/NOSLOTS**.

If a volume range is specified, you can specify a slot range, and each volume will be placed in each slot in order. For a single volume, specify a single slot. This qualifier requires the right MDMS_SET_PROTECTED.

/SPACES=(*range*[,...])
/NOSPACES

This qualifier specifies the non-jukebox space in the specified location that the volume resides in. If the volume does not reside in a location space, specify **/NOSPACES**. Use a space range only when creating multiple volumes—each volume will be placed in each space in order. For a single volume, specify a single space.

A single space is a string containing a maximum of 8 alpha numeric characters.

A valid range consists of the same number of characters in both the first and last values, plus a "-" (hyphen) character to separate the first and last values. In a range, numbers are incremented from 0-9, and alphabetic characters are incremented from A-Z (upper case) or a-z (lower case).

A space range can contain up to 1000 spaces (i.e. the difference between the first value and last value should be less than 1000). For example, 1000-5000 would be an invalid range, as the range here is 4000.

A comma-separated list of values can also be specified. This can be used for multiple volumes. For a single volume, the characters after the comma are ignored by MDMS.

The following table gives examples of valid ranges:

Non-Numeric Range	Numeric Range
A4-A6	10-15
aac-bbb	123-456
	1000-1500
	10001-10020
abc-def (Range is more than 1000 spaces)	1000-5000
abc-defg (Number of characters differ)	

/STATE=*keyword*

This is a protected field that should be modified only to recover on error. Use the **ALLOCATE VOLUME** or **DEALLOCATE VOLUME** command to manipulate the state field under normal operation. This qualifier allows you to modify the state of the volume. This is normally set up by MDMS and manual modification is not recommended. The keyword values are:

ALLOCATED

FREE
TRANSITION
UNAVAILABLE
UNINITIALIZED

This qualifier requires the right MDMS_SET_PROTECTED.

/TIMES_CLEANED=number

This qualifier allows you to specify the number of times the volume has been cleaned. The default is zero.

/TRANSITION_TIME=delta-time
/NOTTRANSITION_TIME

The **/TRANSITION_TIME** qualifier specifies that the volume enters the TRANSITION state when the scratch date is reached, and is to stay in the TRANSITION state for the specified delta time. When the transition time has expired, the volume enters the FREE state. The **/NOTTRANSITION_TIME** qualifier specifies that the volume enters the FREE state directly at the scratch date.

/UNAVAILABLE

Puts the volume in the UNAVAILABLE state. The previous state is retained for when the volume is made available again.

/USER_NAME=username
/NOUSER_NAME

Specifies the user for the volume. The username can be from 1-31 characters, and must reflect an authorized VMS username. To clear the username, enter **/NOUSER_NAME**.

This qualifier requires the right MDMS_SET_PROTECTED.

/VOLSET
/NOVOLSET (Default)

This qualifier specifies that all changes apply to the entire volume set. By default, attributes apply to a single volume or volume range specified. Exceptions to this are **SET VOLUME/ RETAIN** and **SET VOLUME/RELEASE**, which always act on a volume set.

Examples

The following command modifies the on-site location of volume ABC001:

```
$ MDMS SET VOLUME ABC001 /ONSITE=(LOCATION=SHELF2) -  
_ $ /SPACES=AA4
```

The following command modifies the jukebox slot location for volume SFR024 to slot 24:

```
$ MDMS SET VOLUME SFR024 /SLOTS=24
```

The following command modifies the magazine and slot definition for the seven volume records HS0001-HS0007:

```
$ MDMS SET VOLUME HS0001-HS0007 /MAGAZINE=TX877B -
```

\$ /SLOTS=(0-6)

MDMS SHOW ARCHIVE

MDMS SHOW ARCHIVE — The **MDMS SHOW ARCHIVE** command displays information about selected archives or all archives defined in the MDMS database.

Format

MDMS SHOW ARCHIVE [*archive-name*[,...]]

Equivalent ABS Command: **SHOW STORAGE_CLASS**

Parameters

archive-name

Specify the archive names. All archives are displayed if the *archive-name* parameter is omitted. Wildcard characters are allowed in the *archive-name*, where * represents a variable-length wildcard, and % represents a single-character wildcard.

The maximum length of the archive name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of archive names may be entered.

Description

The **MDMS SHOW ARCHIVE** command displays information about specified archives in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/BRIEF

Display archive information in the brief 1-line format. This is the default display when an archive name is not specified.

/FULL

Display archive information in the full multi-line format. All attributes and status fields are displayed. This is the default display when an archive name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the archive information. The default is SYSS\$OUTPUT.

/SYMBOLS [=prefix]

Stores selected archive information in process symbols. If prefix is not specified, the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single archive:

MDMS_INQ_ARCHIVE_NAME	MDMS_INQ_ARCHIVE_TYPE
MDMS_INQ_CATALOG_NAME	MDMS_INQ_CONSOLIDATION_INTERVAL
MDMS_INQ_CONSOLIDATION_SAVESETS	MDMS_INQ_CONSOLIDATION_VOLUMES
MDMS_INQ_DESCRIPTION	MDMS_INQ_DESTINATION
MDMS_INQ_DRIVES	MDMS_INQ_EXPIRATION_DATE
MDMS_INQ_LOCATION_NAME	MDMS_INQ_MAXIMUM_SAVES
MDMS_INQ_MEDIA_TYPE	MDMS_INQ_OWNER_NAME
MDMS_INQ_POOL_NAME	MDMS_INQ_RETENTION_DAYS
MDMS_INQ_VOLUME_SET	

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing archive JIMS_ARCHIVE:

```
$ MDMS SHOW ARCHIVE JIMS_ARCHIVE /FULL
```

The following command displays a brief listing of all archive information and sends the output into file ARCHIVES.DAT:

```
$ MDMS SHOW ARCHIVE /OUTPUT=ARCHIVES.DAT
```

MDMS SHOW CATALOG

MDMS SHOW CATALOG — Displays information about selected catalogs or all catalogs defined in the ABS database.

Format 1

```
MDMS SHOW CATALOG [catalog-name] [, ...]
```

Shows the attributes of a catalog.

Equivalent ABS Command: **RUN SYS\$SYSTEM:ABS\$CATALOG_OBJECT**

Format 2

```
MDMS SHOW CATALOG [catalog-name] [, ...] /SAVE
```

Shows save entries in a catalog.

Equivalent ABS Command: **ABS REPORT SAVE_LOG**

Format 3

```
MDMS SHOW CATALOG [catalog-name] [, ...] /FILES
```

Show file entries in a catalog.

Equivalent ABS Command: **ABS LOOKUP**

Parameters

catalog-name

Specify the catalog names. All catalogs are displayed if the *catalog-name* parameter is omitted. Wildcard characters are allowed in the *catalog-name*, where * represents a variable-length wildcard and % represents a single character wildcard.

The maximum length of the catalog name is 26 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of catalog names may be entered.

Description

The **MDMS SHOW CATALOG** command displays information about specified catalogs in the MDMS database.

A UID (Unique Identifier) is displayed in case the save, archive, or environment name of the original object is no longer available to retrieve its name.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

/SYMBOLS are only defined on a show of a single catalog or a single catalog item.

When doing a show of ORACLE_DB type catalogs, see the notes for **/INCLUDE** and **/SAVES**.

Not supported for SLS type catalogs.

Qualifiers

/BRIEF

Display catalog information in the brief 1-line format. This is the default display when a catalog name is not specified and neither **/FILES** nor **/SAVES** is specified. Whenever **/FILES** or **/SAVES** is present, the default is brief display.

/FILES[=(filename[,...])]

Display information about archived files. Wildcards are supported anywhere in the filename.

If a filename parameter is specified, the **/DATA_SELECT_TYPE** qualifier is required.

See **/FILES**-specific qualifiers under *the section called “Files Qualifiers”*.

Not supported for SLS type catalogs.

/FULL

Display catalog information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a catalog name is specified.

/NODES[=(node-name[,...])]

The **/NODES** qualifier specifies on which node the catalog is located.

The default is the node on which the **SHOW CATALOG** command is entered.

/OUTPUT=file-spec

Specifies the output file in which to send the catalog information. The default is SYS\$OUTPUT.

/SAVES[=(save-name[,...])]

Displays information about save operations. Wildcards are supported in the *save-name*. The *save-name* can also be a UID string in case the save object has been deleted and only the UID is known.

Note

The *save-name* has no meaning for ORACLE_DB type catalogs because no save objects are involved in saving Oracle database. Specifying a *save-name* for ORACLE_DB type catalogs will not display any catalog entry.

See **/SAVES**-specific qualifiers under *the section called “Saves Qualifiers”*.

/SYMBOLS

Stores selected catalog information in process symbols. By default, there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are created:

MDMS_INQ_CATALOG_NAME
MDMS_INQ_CATALOG_NODE
MDMS_INQ_CATALOG_TYPE
MDMS_INQ_DIRECTORY
MDMS_INQ_STAGING_ENABLED

Symbols are not available for access control.

Saves Qualifiers

/SAVES/ARCHIVE=archive-name

Displays entries for the specified archive. By default, entries for all archives are displayed.

/SAVES/BEFORE[=*date*]

Display save entries with a date archived before this time. You can specify time as an absolute time, as a combination of absolute and delta times, or as one of the following keywords: TODAY (default), TOMORROW, or YESTERDAY. Specify **/ALL** with the **/BEFORE** qualifier to list all entries of a file in reverse chronological order.

If you do not specify any date selection, files are not selected based on a date. If you do not specify any date, the current date at midnight is used.

/SAVES/INCLUDE=(*include-specification*[,...])

The **/INCLUDE** qualifier specifies the exact include string used in a SAVE operation. Only entries with matching include specifications are selected.

Note

If you are showing information for ORACLE_DB type catalogs, the *include-specification* will be the database name

/SAVES/PIECE_NAME=*piece-name*

Displays information about an Oracle piece name. Applicable to ORACLE_DB type catalogs only.

/SAVES/SINCE[=*time*]

Display save entries with a date archived since this time. You can specify time as an absolute time, as a combination of absolute and delta times, or as one of the following keywords: TODAY (default), TOMORROW, or YESTERDAY. Specify **/ALL** with the **/SINCE** qualifier to list all entries of a file in chronological order. If used together with **/BEFORE** the all entries are listed in reverse chronological order.

If you do not specify any date selection, files are not selected based on a date. If you do not specify any date, the current date at midnight is used.

/SAVES/SYMBOLS/FULL

Stores selected catalog information in process symbols. By default, there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are created:

MDMS_INQ_ARCHIVE_NAME	MDMS_INQ_ARCHIVE_TYPE
MDMS_INQ_BLOCK_SIZE	MDMS_INQ_DATABASE
MDMS_INQ_DATE_ARCHIVED	MDMS_INQ_ENVIRONMENT_NAME
MDMS_INQ_INCLUDE	MDMS_INQ_OBJECT_ENTRIES
MDMS_INQ_OWNER	MDMS_INQ_PIECE_NAME
MDMS_INQ_SAVE_NAME	MDMS_INQ_SAVE_TYPE
MDMS_INQ_SAVESET_FORMAT	MDMS_INQ_SAVESET_NAME
MDMS_INQ_SAVESET_LOCATION	MDMS_INQ_SAVESET_POSITION
MDMS_INQ_SEVERITY	MDMS_INQ_SOURCE_NODE

MDMS_INQ_STATUS

Note

The **/FULL** qualifier is required for **MDMS SHOW CATALOG/SAVES/SYMBOLS** as MDMS does not generate symbols for brief display.

/SAVES/VOLUMES=(*volume-set*[,...])

Displays information about entries saved on the specified volume sets. By default, entries for all volume sets are shown.

To find the volume set name for a given volume use **MDMS SHOW VOLUME/VOLSET**. The first volume is the name of the volume set.

Files Qualifiers

/FILES/ALL

Display all entries of files. If not specified, only the most recently archived entry is displayed.

/FILES/BEFORE[=*date*]

Display file entries with a date archived before this time. If no time is given, the current date and time is used.

/FILES/DATA_SELECT_TYPE=*keyword*

The **/DATA_SELECT_TYPE** qualifier specifies the type of data to display. This qualifier is required if **/FILES** is used with a filename. Select one of the following options:

- VMS_FILES
- VMS_SAVESET

If not specified, all entries are displayed.

/FILES/SINCE[=*date*]

Display file entries with a date archived since this time. If no time is given the current date and time is used.

/FILES/SYMBOLS/FULL

Stores selected catalog information in process symbols. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are created:

MDMS_INQ_CREATION_DATE	MDMS_INQ_DATA_SELECT_TYPE
MDMS_INQ_DATE_ARCHIVED	MDMS_INQ_EXPIRATION_DATE
MDMS_INQ_FILENAME	MDMS_INQ_OWNER
MDMS_INQ_REVISION_DATE	MDMS_INQ_SAVE_TYPE
MDMS_INQ_SAVESET_NAME	MDMS_INQ_SAVESET_LOCATION

MDMS_INQ_SAVESET_SECTION

MDMS_INQ_SEVERITY

MDMS_INQ_SOURCE_NODE

MDMS_INQ_STATUS

Note

The **/FULL** qualifier is required for **MDMS SHOW CATALOG/FILES/SYMBOLS** as MDMS does not generate symbols for brief display.

Examples

The following command displays a full listing catalog MY_CATALOG:

```
$ MDMS SHOW CATALOG MY_CATALOG /FULL
```

The following command displays a brief listing of all catalog information and sends the output into file CATALOGS.DAT:

```
$ MDMS SHOW CATALOG /OUTPUT=CATALOGS.DAT
```

MDMS SHOW DOMAIN

MDMS SHOW DOMAIN — Displays information about the MDMS domain.

Format

```
MDMS SHOW DOMAIN
```

Equivalent STORAGE Command: None.

Parameters

None.

Description

The **MDMS SHOW DOMAIN** command displays information about the MDMS domain.

Privileges

The request requires MDMS_SHOW_ALL.

If the **/FULL** qualifier is specified, MDMS_SHOW_RIGHTS is also required.

Restrictions

None.

Qualifiers

/FULL

Shows the rights assignments in the display.

This qualifier requires the right MDMS_SHOW_RIGHTS.

/OUTPUT=*file-spec*

Specifies the output file in which to send the domain information. The default is SYSS\$OUTPUT.

/SYMBOLS[=*prefix*]

Stores selected domain information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are created:

MDMS_INQ_CHECK_ACCESS	MDMS_INQ_DATABASE_DATE
MDMS_INQ_DEALLOCATE_STATE	MDMS_INQ_DESCRIPTION
MDMS_INQ_LAST_UPDATED	MDMS_INQ_MAIL
MDMS_INQ_MAX_SCRATCH_TIME	MDMS_INQ_MEDIA_TYPE
MDMS_INQ_NODE_NAME	MDMS_INQ_OFFSITE_LOCATION
MDMS_INQ_ONSITE_LOCATION	MDMS_INQ_OPCOM_CLASS
MDMS_INQ_PROTECTION	MDMS_INQ_RELAXED_ACCESS
MDMS_INQ_REQUEST_ID	MDMS_INQ_SCHEDULER_TYPE
MDMS_INQ_SCRATCH_TIME	MDMS_INQ_TRANSITION_TIME

Symbols are not available for access control, domain rights and for holiday definitions.

Examples

The following command displays a brief listing of the MDMS domain:

```
$ MDMS SHOW DOMAIN
```

The following command outputs a full listing of the MDMS domain to file DOMAIN.DAT:

```
$ MDMS SHOW DOMAIN /OUTPUT=DOMAIN.DAT
```

The following command displays a full listing of the MDMS domain including the rights attributes which are normally hidden:

```
$ MDMS SHOW DOMAIN/FULL
```

MDMS SHOW DRIVE

MDMS SHOW DRIVE — Displays information about specified drives or all drives defined in the MDMS database.

Format

```
MDMS SHOW DRIVE [drive-name[,...]]
```

Equivalent STORAGE Command: None.

Parameters

drive-name

Specifies the names of the drives. Specify a drive name or a logical name. All drives are displayed if the *drive-name* parameter is omitted. Wildcard characters are allowed in the drive-name, where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the drive name or logical name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of drive names may be entered.

Description

The **MDMS SHOW DRIVE** command displays information about specified drives or all drives defined in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

None.

Qualifiers

/BRIEF

Display drive information in the brief 1-line format. This is the default display when a drive name is not specified.

/CHECK

/NOCHECK (Default)

When the **/CHECK** qualifier is specified, the drive is physically accessed to determine its availability and its state.

If the state varies from what is stored in the database, the state is updated (and associated volume records are also updated) and the new values are displayed. If the check cannot be performed, the original database settings are displayed with a warning message. With the default **/NOCHECK**, no physical checking is performed, and the current database contents are displayed.

This qualifiers requires the right MDMS_SET_OWN.

/FULL

Display drive information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a drive name is specified.

/OUTPUT=file-spec

Specifies the output file in which to send the drive information. The default is SYS\$OUTPUT.

/SYMBOLS[=*prefix*]

Stores selected drive information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single drive:

MDMS_INQ_ACCESS	MDMS_INQ_ALLOCATED
MDMS_INQ_ALLOC_DEVICE_NAME	MDMS_INQ_ALLOC_NODE_NAME
MDMS_INQ_ALLOC_OWNER_UIC	MDMS_INQ_ALLOC_PROCESS_ID
MDMS_INQ_ALLOC_USER_NAME	MDMS_INQ_AUTOMATIC_REPLY
MDMS_INQ_AVAILABLE	MDMS_INQ_DESCRIPTION
MDMS_INQ_DEVICE_NAME	MDMS_INQ_DISABLED
MDMS_INQ_DRIVE_NAME	MDMS_INQ_DRIVE_NUMBER
MDMS_INQ_GROUPS	MDMS_INQ_JUKEBOX_NAME
MDMS_INQ_LOAD_VOLUME_ID	MDMS_INQ_MEDIA_TYPE
MDMS_INQ_NODES	MDMS_INQ_OWNER_NAME
MDMS_INQ_READONLY_MEDIA_TYPE	MDMS_INQ_SHARED
MDMS_INQ_STACKER	MDMS_INQ_STATE
MDMS_INQ_VOLUME_ID	

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing of drive information for drive \$1\$MUA5 to SYS
\$OUTPUT:

```
$ MDMS SHOW DRIVE $1$MUA5
```

The following command displays a full listing of drive information for drive \$1\$MUA5 and puts the output into file DRIVE.DAT:

```
$ MDMS SHOW DRIVE /FULL $1$MUA5 /OUTPUT=DRIVES.DAT
```

MDMS SHOW ENVIRONMENT

MDMS SHOW ENVIRONMENT — Displays information about selected environments or all environments defined in the MDMS database.

Format

```
MDMS SHOW ENVIRONMENT [environment-name[,...]]
```

Equivalent ABS Command: **SHOW ENVIRONMENT**

Parameters

environment-name

Specify the environment names. All environments are displayed if the *environment-name* parameter is omitted. Wildcard characters are allowed in the *environment-name*, where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the environment name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of environment names may be entered.

Description

The **MDMS SHOW ENVIRONMENT** command displays information about specified environments in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/BRIEF

Display environment information in the brief 1-line format. This is the default display when an environment name is not specified.

/FULL

Display environment information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a environment name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the environment information. The default is SYS\$OUTPUT.

/SYMBOLS

Stores selected environment information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single environment:

MDMS_INQ_ACTION

MDMS_INQ_ASSIST

MDMS_INQ_COMPRESSION	MDMS_INQ_DATA_SAFETY
MDMS_INQ_DESCRIPTION	MDMS_INQ_DRIVE_COUNT
MDMS_INQ_ENVIRONMENT_NAME	MDMS_INQ_EPILOGUE
MDMS_INQ_INTERVAL	MDMS_INQ_LINKS_ONLY
MDMS_INQ_LISTING_OPTION	MDMS_INQ_LOCK
MDMS_INQ_OWNER_NAME	MDMS_INQ_PROFILE_CLUSTER
MDMS_INQ_PROFILE_NODE	MDMS_INQ_PROFILE_PRIVILEGES
MDMS_INQ_PROFILE_RIGHTS	MDMS_INQ_PROFILE_USER
MDMS_INQ_PROLOGUE	MDMS_INQ_RETRY_LIMIT
MDMS_INQ_SPAN_FILESYSTEMS	

Symbols are not available for access control and notification.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing environment `BACKUP_ENV`:

```
$ MDMS SHOW ENVIRONMENT BACKUP_ENV /FULL
```

The following command displays a brief listing of all environment information and sends the output into file `ENVIRONMENTS.DAT`:

```
$ MDMS SHOW ENVIRONMENT /OUTPUT=ENVIRONMENTS.DAT
```

MDMS SHOW GROUP

MDMS SHOW GROUP — Displays information about selected groups or all groups defined in the MDMS database.

Format

```
MDMS SHOW GROUP [group-name[,...]]
```

Equivalent STORAGE Command: None.

Parameters

group-name

Specify the group names. All groups are displayed if the *group-name* parameter is omitted. Wildcard characters are allowed in the *group-name*, where *** represents a variable-length wildcard, and *%* represents a single character wildcard.

The maximum length of the group name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of group names may be entered.

Description

The **MDMS SHOW GROUP** command displays information about specified groups in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

None.

Qualifiers

/BRIEF

Display group information in the brief 1-line format. This is the default display when a group name is not specified.

/FULL

Display group information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a group name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the group information. The default is SYS\$OUTPUT.

/SYMBOLS[=*prefix***]**

Stores selected group information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single group:

MDMS_INQ_DESCRIPTION	MDMS_INQ_GROUP_NAME
MDMS_INQ_NODES	MDMS_INQ_OWNER_NAME

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing group HOSER:

```
$ MDMS SHOW GROUP HOSER /FULL
```

The following command displays a brief listing of all group information and sends the output into file GROUPS.DAT:

```
$ MDMS SHOW GROUP /OUTPUT=GROUPS.DAT
```

MDMS SHOW JUKEBOX

MDMS SHOW JUKEBOX — Displays information about specified jukeboxes, selected jukeboxes, or all jukeboxes defined in the MDMS database.

Format

```
MDMS SHOW JUKEBOX [jukebox-name[,...]]
```

Equivalent STORAGE Command: None.

Parameters

jukebox-name

Specifies the names of the jukeboxes. All jukeboxes are displayed if the *jukebox-name* parameter is omitted. Wildcard characters are allowed in the *jukebox-name*, where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the jukebox name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of jukebox names may be entered.

Description

The **MDMS SHOW JUKEBOX** command displays information about specified jukeboxes or all jukeboxes defined in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

None.

Qualifiers

/BRIEF

Display jukebox information in the brief 1-line format. This is the default display when a jukebox name is not specified.

/CONTENTS

Display information about the contents of the jukebox—drives, slots, volumes, and magazines.

/FULL

Display jukebox information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a jukebox name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the jukebox information. The default is SYS\$OUTPUT.

/SYMBOLS[=*prefix*]

Stores selected jukebox information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single jukebox:

MDMS_INQ_ACCESS	MDMS_INQ_ACS_ID
MDMS_INQ_AUTOMATIC_REPLY	MDMS_INQ_CAP_SIZE
MDMS_INQ_CONTROL	MDMS_INQ_DESCRIPTION
MDMS_INQ_DISABLED	MDMS_INQ_FREE_VOLUMES
MDMS_INQ_GROUPS	MDMS_INQ_JUKEBOX_NAME
MDMS_INQ_LIBRARY	MDMS_INQ_LOCATION_NAME
MDMS_INQ_LSM_ID	MDMS_INQ_NODES
MDMS_INQ_OWNER_NAME	MDMS_INQ_ROBOT
MDMS_INQ_SLOT_COUNT	MDMS_INQ_STATE
MDMS_INQ_THRESHOLD	MDMS_INQ_TOPOLOGY_FACE
MDMS_INQ_TOPOLOGY_LEVEL	MDMS_INQ_TOPOLOGY_SLOT
MDMS_INQ_TOPOLOGY_TOWER	MDMS_INQ_USAGE

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing of jukebox information for jukebox JUKE_1:

```
$ MDMS SHOW JUKEBOX JUKE_1
```

The following command displays a full listing for all jukeboxes, which is stored in file JUKE_LIST.DAT instead of SYS\$OUTPUT:

```
$ MDMS SHOW JUKEBOX /FULL /OUTPUT=JUKE_LIST.DAT
```

MDMS SHOW LOCATION

MDMS SHOW LOCATION — Displays information about specified locations or all locations defined in the MDMS database.

Format

```
MDMS SHOW LOCATION [location[,...]]
```

Equivalent STORAGE Command: None.

Parameters

location

Specifies the name of the location. All locations are displayed if the *location* parameter is omitted. Wildcard characters are allowed in the *location* where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the location is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of locations may be entered.

Description

The **MDMS SHOW LOCATION** command displays information about specified locations or all locations defined in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

None.

Qualifiers

/BRIEF

Display location information in the brief 1-line format. This is the default display when a location name is not specified.

/FULL

Display location information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a location name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the location information. The default is SYS\$OUTPUT.

/SYMBOLS[=*prefix***]**

Stores selected location information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single location:

MDMS_INQ_DESCRIPTION

MDMS_INQ_LOCATION_NAME

MDMS_INQ_OWNER_NAME

MDMS_INQ_PARENT_LOCATION

MDMS_INQ_SPACES

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a brief listing of all locations:

```
$ MDMS SHOW LOCATION
```

The following command displays a full listing of location information for locations SHELF1 to SHELF4:

```
$ MDMS SHOW LOCATION SHELF1, SHELF2, SHELF3, SHELF4 /FULL
```

The following command displays a full listing of all location information in file LOC_LIST.DAT:

```
$ MDMS SHOW LOCATION /FULL /OUTPUT=LOC_LIST.DAT
```

MDMS SHOW MAGAZINE

MDMS SHOW MAGAZINE — Displays information about specified magazines, selected magazines, or all magazines defined in the MDMS database.

Format

```
MDMS SHOW MAGAZINE [magazine-name[,...]]
```

Equivalent STORAGE Command: **STORAGE SHOW MAGAZINE**

Parameters

magazine-name

Specifies the names of the magazines. All magazines are displayed if the *magazine-name* parameter is omitted, subject to the **/SCHEDULE** qualifier. Wildcard characters are allowed in the *magazine-name*, where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the magazine name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of magazine names may be entered.

Description

The **MDMS SHOW MAGAZINE** command displays information about specified magazines or all magazines defined in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

None.

Qualifiers

/BRIEF

Display magazine information in the brief 1-line format. This is the default display when a magazine name is not specified.

/CONTENTS

Display information about the contents of the magazine—slots and volumes.

/FULL

Display magazine information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a magazine name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the magazine information. The default is SYS\$OUTPUT.

/SCHEDULE[=*keyword***]**

The **/SCHEDULE** qualifier selects magazines whose off-site or on-site date has "expired" and the magazine is not in the new location. If both dates have expired, the later of the two dates is used. The optional keywords on the **/SCHEDULE** qualifier may be OFFSITE or ONSITE.

If the OFFSITE keyword is used, then only those magazines scheduled to be moved off site are selected. If the ONSITE keyword is used, then only those magazines scheduled to be moved on site are selected. If the keyword is omitted, then the magazines scheduled to be moved on site and off site are selected. Do not specify a magazine name with this qualifier.

/SYMBOLS

Stores selected magazine information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single magazine:

MDMS_INQ_DESCRIPTION	MDMS_INQ_JUKEBOX_NAME
MDMS_INQ_MAGAZINE_NAME	MDMS_INQ_OFFSITE_DATE
MDMS_INQ_OFFSITE_LOCATION	MDMS_INQ_ONSITE_DATE
MDMS_INQ_ONSITE_LOCATION	MDMS_INQ_OWNER_NAME
MDMS_INQ_PLACEMENT	MDMS_INQ_PLACENAME
MDMS_INQ_POSITION	MDMS_INQ_SLOT_COUNT
MDMS_INQ_SPACES	MDMS_INQ_START_SLOT

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a brief listing of all magazines:

```
$ MDMS SHOW MAGAZINE
```

The following command outputs a full listing of magazine information for magazine matching the wildcard PDMAG* to file PDMAG1.DAT:

```
$ MDMS SHOW MAGAZINE PDMAG* /OUTPUT=PDMAG1.DAT
```

The following command outputs a brief listing of all magazines scheduled to be moved on site or off site:

```
$ MDMS SHOW MAGAZINE /SCHEDULE
```

MDMS SHOW MEDIA_TYPE

MDMS SHOW MEDIA_TYPE — Displays information about specified media types or all media types defined in the MDMS database.

Format

```
MDMS SHOW MEDIA_TYPE [media-type[,...]]
```

Equivalent STORAGE Command: None.

Parameters

media-type

Specifies the names of the media types. All media types are displayed if the *media-type* parameter is omitted. Wildcard characters are allowed in the *media-type*, where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the media type name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of media types may be entered.

Description

The **MDMS SHOW MEDIA_TYPE** command displays information about specified media types or all media types defined in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

None.

Qualifiers

/BRIEF

Display media type information in the brief 1-line format. This is the default display when a media type is not specified.

/FULL

Display media type information in the full multi- line format. All attributes and status fields are displayed. This is the default display when a media type is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the media type information. The default is SYS\$OUTPUT.

/SYMBOLS

Stores selected media type information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single media type:

MDMS_INQ_CAPACITY	MDMS_INQ_COMPACTION
MDMS_INQ_DENSITY	MDMS_INQ_DESCRIPTION
MDMS_INQ_MEDIA_TYPE	MDMS_INQ_OWNER_NAME
MDMS_INQ_TAPE_LENGTH	

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing of information for media type TK88K_COMP:

```
$ MDMS SHOW MEDIA_TYPE TK88K_COMP /FULL
```

The following command outputs a full listing of information for all media types to file MEDIA.DAT:

```
$ MDMS SHOW MEDIA_TYPE /FULL /OUTPUT=MEDIA.DAT
```

MDMS SHOW NODE

MDMS SHOW NODE — Displays information about specified nodes or all nodes defined in the MDMS database.

Format

MDMS SHOW NODE [*node-name*[, ...]]

Equivalent STORAGE Command: None.

Parameters

node-name

Specifies the names of the nodes. All nodes are displayed if the *node-name* parameter is omitted. Wildcard characters are allowed in the *node-name*, where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the node name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of node names may be entered.

Description

The **MDMS SHOW NODE** command displays information about specified nodes or all nodes defined in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

None.

Qualifiers

/BRIEF

Display node information in the brief 1-line format. This is the default display when a node name is not specified.

/FULL

Display node information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a node name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the node information. The default is SYS\$OUTPUT.

/SYMBOLS[=*prefix***]**

Stores selected node information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single node:

MDMS_INQ_DATABASE_SERVER	MDMS_INQ_DECNET_FULLNAME
MDMS_INQ_DESCRIPTION	MDMS_INQ_DISABLED
MDMS_INQ_LOCATION_NAME	MDMS_INQ_NODE_NAME
MDMS_INQ_OPCOM_CLASS	MDMS_INQ_OWNER_NAME
MDMS_INQ_TCPIP_FULLNAME	MDMS_INQ_TRANSPORT

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command outputs a full listing of node information for all nodes to file `NODES.DAT`:

```
$ MDMS SHOW NODE /FULL /OUTPUT=NODES.DAT
```

The following command displays a full listing of node information for node `DSORDS`:

```
$ MDMS SHOW NODE DSORDS
```

The following command displays a full listing of nodes matching the wildcard `SIL*`:

```
$ MDMS SHOW NODE SIL*/FULL
```

MDMS SHOW POOL

MDMS SHOW POOL — Displays information about specified pools or all pools defined in the MDMS database.

Format

```
MDMS SHOW POOL [pool-name[,...]]
```

Equivalent STORAGE Command: None.

Parameters

pool-name

Specifies the name of the pool. All pools are displayed if the *pool-name* parameter is omitted. Wildcard characters are allowed in the *pool-name*, where *** represents a variable-length wildcard, and *%* represents a single character wildcard.

The maximum length of the pool name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of pool names may be entered.

Description

The **MDMS SHOW POOL** command displays information about specified pools or all pools defined in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

None.

Qualifiers

/BRIEF

Display pool information in the brief 1-line format. This is the default display when a pool name is not specified.

/FULL

Display pool information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a pool name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the pool information. The default is SYS\$OUTPUT.

/SYMBOLS[=*prefix***]**

Stores selected pool information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single pool:

MDMS_INQ_AUTHORIZED_USERS	MDMS_INQ_DEFAULT_USERS
MDMS_INQ_DESCRIPTION	MDMS_INQ_FREE_VOLUMES
MDMS_INQ_OWNER_NAME	MDMS_INQ_POOL_NAME
MDMS_INQ_THRESHOLD	

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing of pool information for pool TEST_POOL:

```
$ MDMS SHOW POOL TEST_POOL /FULL
```

MDMS SHOW REQUEST

MDMS SHOW REQUEST — Displays information about all requests currently active on the database server node, which includes virtually all requests in the domain.

Format

MDMS SHOW REQUEST [*request-id*]

Equivalent STORAGE Command: None.

Parameters

request-id

Specifies an identifier to show a specific request. You can obtain a list of request IDs by issuing a **SHOW REQUEST** command without a request ID.

Description

The **MDMS SHOW REQUESTS** command displays information about requests currently active in the domain. Certain recently-completed requests may also be shown. If no *request-id* is specified, all requests are shown subject to privilege. In addition, requests from a specific user can be shown.

Privileges

The request requires MDMS_SHOW_ALL, MDMS_SHOW_POOL, or MDMS_SHOW_OWN.

If the user has only MDMS_SHOW_OWN or MDMS_SHOW_POOL, only requests issued by the user are shown. To see requests of other users, MDMS_SHOW_ALL is required.

Restrictions

None.

Qualifiers

/BRIEF

Displays requests information in the brief 1-line format. This is the default display if no request ID is specified.

/DEBUG

Displays addresses of certain objects for debugging purposes.

/FULL

Displays request information in the full multi-line format. This is the default display if a request ID is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the requests information. The default is SYS\$OUTPUT.

/USER_NAME=*username*

Restricts the request list to those issued by the specified user.

Examples

The following command displays a brief listing of all requests active in the domain:

```
$ MDMS SHOW REQUESTS
```

The following command displays a full listing of all requests active in the domain:

```
$ MDMS SHOW REQUEST/FULL
```

The following command displays a full listing of request identifier 45:

```
$ MDMS SHOW REQUEST 45
```

The following command displays a brief listing of all requests issued by user FROEHLIN:

```
$ MDMS SHOW REQUEST/USER_NAME=FROEHLIN
```

MDMS SHOW RESTORE

MDMS SHOW RESTORE — Displays information about selected restores or all restores defined in the MDMS database.

Format

```
MDMS SHOW RESTORE [restore-name[,...]]
```

Equivalent ABS Command: **SHOW STORAGE_CLASS**

Parameters

restore-name

Specify the restore names. All restores are displayed if the *restore-name* parameter is omitted. Wildcard characters are allowed in the *restore-name*, where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the restore name is 39 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of restore names may be entered.

Description

The **MDMS SHOW RESTORE** command displays information about specified restores in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL. In addition, the **/PROFILE** qualifier requires MDMS_SHOW_RIGHTS.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/BRIEF

Display restore information in the brief 1-line format. This is the default display when a restore name is not specified.

/FULL

Display restore information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a restore name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the restore information. The default is SYS\$OUTPUT.

/PROFILE

Outputs the user profile stored in the restore request. By default the user profile is not displayed.

This qualifier requires MDMS_SHOW_RIGHTS.

/SELECTIONS

Outputs the details of all selections in the restore. By default, only selected information about the default selection that is settable in the restore command is shown. This qualifier is not supported for wildcard show requests.

/SYMBOLS[=*prefix***]**

Stores selected restore information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single restore:

MDMS_INQ_ARCHIVE_NAME	MDMS_INQ_BASE_DATE
MDMS_INQ_BEFORE_DATE	MDMS_INQ_CATALOG_NAME
MDMS_INQ_DATA_SELECT_TYPE	MDMS_INQ_DATE_ARCHIVED
MDMS_INQ_DELETE_INTERVAL	MDMS_INQ_DESCRIPTION
MDMS_INQ_DESTINATION	MDMS_INQ_ENVIRONMENT_NAME
MDMS_INQ_EPILOGUE	MDMS_INQ_EXCLUDE
MDMS_INQ_EXPLICIT_INTERVAL	MDMS_INQ_FREQUENCY
MDMS_INQ_GROUPS	MDMS_INQ_INCLUDE
MDMS_INQ_INCREMENTAL	MDMS_INQ_JOB_NUMBER
MDMS_INQ_NODES	MDMS_INQ_OWNER_NAME
MDMS_INQ_PROFILE_CUSTER	MDMS_INQ_PROFILE_NODE
MDMS_INQ_PROFILE_PRIVILEGES	MDMS_INQ_PROFILE_RIGHTS

MDMS_INQ_PROFILE_USER	MDMS_INQ_PROLOGUE
MDMS_INQ_RESTORE_NAME	MDMS_INQ_SCHEDULE_NAME
MDMS_INQ_SELECTIONS	MDMS_INQ_SEQUENCE_OPTION
MDMS_INQ_SINCE_DATE	MDMS_INQ_SKIP_TIME
MDMS_INQ_SOURCE_NODE	MDMS_INQ_START_DATE
MDMS_INQ_TRANSACTION_STATUS	

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing restore CHRIS_RESTORE:

```
$ MDMS SHOW RESTORE CHRIS_RESTORE /FULL
```

The following command displays a brief listing of all restore information and sends the output into file RESTORES.DAT:

```
$ MDMS SHOW RESTORE /OUTPUT=RESTORES.DAT
```

MDMS SHOW SAVE

MDMS SHOW SAVE — Displays information about selected saves or all saves defined in the MDMS database.

Format

```
MDMS SHOW SAVE [save-name] [, ...]
```

Equivalent ABS Command: **SHOW STORAGE_CLASS**

Parameters

save-name

Specify the save names. All saves are displayed if the *save-name* parameter is omitted. Wildcard characters are allowed in the *save-name*, where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the save name is 39 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of save names may be entered.

Description

The **MDMS SHOW SAVE** command displays information about specified saves in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL. In addition, the **/PROFILE** qualifier requires MDMS_SHOW_RIGHTS.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/BRIEF

Display save information in the brief 1-line format. This is the default display when a save name is not specified.

/FULL

Display save information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a save name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the save information. The default is SYS\$OUTPUT.

/PROFILE

Outputs the user profile stored in the save request. By default the user profile is not displayed.

This qualifier requires MDMS_SHOW_RIGHTS.

/SELECTIONS

Outputs the details of all selections in the save. By default, only selected information about the default selection that is settable in the save command is shown. This qualifier is not supported for wildcard show requests.

/SYMBOLS[=*prefix***]**

Stores selected save information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single save:

MDMS_INQ_ARCHIVE_NAME	MDMS_INQ_BASE_DATE
MDMS_INQ_DATA_SELECT_TYPE	MDMS_INQ_DELETE_INTERVAL
MDMS_INQ_DESCRIPTION	MDMS_INQ_ENVIRONMENT_NAME
MDMS_INQ_EPILOGUE	MDMS_INQ_EXCLUDE
MDMS_INQ_EXPLICIT_INTERVAL	MDMS_INQ_FREQUENCY
MDMS_INQ_GROUPS	MDMS_INQ_INCLUDE

MDMS_INQ_INCREMENTAL	MDMS_INQ_JOB_NUMBER
MDMS_INQ_NODES	MDMS_INQ_OWNER_NAME
MDMS_INQ_PROFILE_CUSTER	MDMS_INQ_PROFILE_NODE
MDMS_INQ_PROFILE_PRIVILEGES	MDMS_INQ_PROFILE_RIGHTS
MDMS_INQ_PROFILE_USER	MDMS_INQ_PROLOGUE
MDMS_INQ_SAVE_NAME	MDMS_INQ_SCHEDULE_NAME
MDMS_INQ_SELECTIONS	MDMS_INQ_SEQUENCE_OPTION
MDMS_INQ_SKIP_TIME	MDMS_INQ_SOURCE_NODE
MDMS_INQ_START_DATE	MDMS_INQ_TRANSACTION_STATUS

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing save MARTYS_SAVE:

```
$ MDMS SHOW SAVE MARTYS_SAVE /FULL
```

The following command displays a brief listing of all save information and sends the output into file SAVES.DAT:

```
$ MDMS SHOW SAVE /OUTPUT=SAVES.DAT
```

MDMS SHOW SCHEDULE

MDMS SHOW SCHEDULE — Displays information about selected schedules or all schedules defined in the MDMS database.

Format

```
MDMS SHOW SCHEDULE [schedule-name[,...]]
```

Equivalent ABS Command: **SHOW STORAGE_CLASS**

Parameters

schedule-name

Specify the schedule names. All schedules are displayed if the *schedule-name* parameter is omitted. Wildcard characters are allowed in the *schedule-name*, where * represents a variable-length wildcard, and % represents a single character wildcard.

The maximum length of the schedule name is 63 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

A comma-separated list of schedule names may be entered.

Description

The **MDMS SHOW SCHEDULE** command displays information about specified schedules in the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

None.

Qualifiers

/BRIEF

Display schedule information in the brief 1-line format. This is the default display when a schedule name is not specified.

/FULL

Display schedule information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a schedule name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the schedule information. The default is SYS\$OUTPUT.

/SYMBOLS[=*prefix***]**

Stores selected schedule information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single schedule:

MDMS_INQ_AFTER_SCHEDULE_NAME	MDMS_INQ_AFTER_SCHEDULE_OPTION
MDMS_INQ_COMMAND	MDMS_INQ_DATES
MDMS_INQ_DAYS	MDMS_INQ_DESCRIPTION
MDMS_INQ_EXCLUDE	MDMS_INQ_INCLUDE
MDMS_INQ_LAST_START_DATE	MDMS_INQ_MONTHS
MDMS_INQ_NEXT_START_DATE	MDMS_INQ_OWNER_NAME
MDMS_INQ_SCHEDULE_NAME	MDMS_INQ_TIMES
MDMS_INQ_WEEKS	

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing schedule `RICHS_SCHEDULE`:

```
$ MDMS SHOW SCHEDULE RICHS_SCHEDULE /FULL
```

The following command displays a brief listing of all schedule information and sends the output into file `SCHEDULES.DAT`:

```
$ MDMS SHOW SCHEDULE /OUTPUT=SCHEDULES.DAT
```

MDMS SHOW SELECTION

MDMS SHOW SELECTION — Displays information about selected selections or all selections defined in the MDMS database.

Format

```
MDMS SHOW SELECTION [selection-name[,...]]
```

Equivalent ABS Command: **SHOW STORAGE_CLASS**

Parameters

selection-name

Specify the selection names. All selections are displayed if the *selection-name* parameter is omitted. Wildcard characters are allowed in the *selection-name*, where `*` represents a variable-length wildcard, and `%` represents a single character wildcard.

The maximum length of the selection name is 63 characters. Valid characters are "\$,-,0-9,AZ,_, a-z".

A comma-separated list of selection names may be entered.

Description

The **MDMS SHOW SELECTION** command displays information about specified selections in the MDMS database.

Privileges

The request requires `MDMS_SHOW_ALL`.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/BRIEF

Display selection information in the brief 1-line format. This is the default display when a selection name is not specified.

/FULL

Display selection information in the full multi-line format. All attributes and status fields are displayed. This is the default display when a selection name is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the selection information. The default is SYS\$OUTPUT.

/SYMBOLS[=*prefix*]

Stores selected selection information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single selection:

MDMS_INQ_AGENT_QUALIFIERS	MDMS_INQ_BEFORE_DATE
MDMS_INQ_CONFLICT_OPTIONS	MDMS_INQ_DATA_SELECT_TYPE
MDMS_INQ_DATE_TYPE	MDMS_INQ_DESCRIPTION
MDMS_INQ_EXCLUDE	MDMS_INQ_INCLUDE
MDMS_INQ_OWNER_NAME	MDMS_INQ_SELECTION_NAME
MDMS_INQ_SINCE_DATE	MDMS_INQ_SOURCE_NODE

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

Examples

The following command displays a full listing selection JOHANNAS_SELECTION:

```
$ MDMS SHOW SELECTION JOHANNAS_SELECTION /FULL
```

The following command displays a brief listing of all selection information and sends the output into file SELECTIONS.DAT:

```
$ MDMS SHOW SELECTION /OUTPUT=SELECTIONS.DAT
```

MDMS SHOW SERVER

MDMS SHOW SERVER — Displays information about the local configuration of the MDMS server on the specified nodes.

Format

```
MDMS SHOW SERVER
```

Equivalent STORAGE Command: None.

Parameters

None.

Description

The **MDMS SHOW SERVER** command displays information about the local configuration of the MDMS server on the specified nodes. The information is derived from the specified nodes local startup configuration files rather than the MDMS database.

Privileges

The request requires MDMS_SHOW_ALL, MDMS_SHOW_POOL, or MDMS_SHOW_OWN.

Restrictions

None.

Qualifiers

/NODES=(*node-name*[,...])

Displays information about the server on the specified nodes. The node name is either a DECnet Phase IV node name, a DECnet-Plus node name, or a TCP/IP node name. A TCP/IP node name can include a port number range. If no port number range is specified the default is 2501-2510.

/OUTPUT=*file-spec*

Specifies the output file in which to send the version information. The default is SYS\$OUTPUT.

/SYMBOLS[=*prefix*]

Stores selected domain information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. By default there is no output on the command. If you wish output with the symbols, use the **/OUTPUT** qualifier. The following symbols are created:

MDMS_INQ_DATABASE_ACCESS	MDMS_INQ_DATABASE_LOCATION
MDMS_INQ_DECNET_FULLNAME	MDMS_INQ_LOGFILE
MDMS_INQ_SERVER_NODE_NAME	MDMS_INQ_SERVER_SEARCH_LIST
MDMS_INQ_SERVER_VERSION	MDMS_INQ_TCPIP_FULLNAME
MDMS_INQ_TRANSPORTS	

Examples

The following command displays MDMS version information on the current node:

```
$ MDMS SHOW SERVER
```

The following command displays local MDMS server information for nodes COOKIE and DSORDS:

```
$ MDMS SHOW SERVER /NODES=(COOKIE, DSORDS)
```

The following command displays information about servers on DECnet node `SENILE`, DECnet-Plus node `PARANOID`, and TCP/IP node `DISORDERS`:

```
$ MDMS SHOW SERVER -  
_ $ /NODES=(SENILE,CORP:.DOM.PARANOID, -  
_ $ DISORDERS.DOM.CORP:3001-3010)
```

MDMS SHOW VERSION

MDMS SHOW VERSION — Displays information about the version of various MDMS components on the specified nodes.

Format

`MDMS SHOW VERSION`

Equivalent `STORAGE` Command: None.

Parameters

None.

Description

The **MDMS SHOW VERSION** command displays information about the version of various MDMS components on the specified nodes.

Privileges

The request requires `MDMS_SHOW_ALL`, `MDMS_SHOW_POOL`, or `MDMS_SHOW_OWN`.

Restrictions

None.

Qualifiers

/NODES=(*node-name*[,...])

Displays information about this server. The node name is either a DECnet Phase IV node name, a DECnet-Plus node name, or a TCP/IP node name. A TCP/IP node name can include a port number range. If no port number range is specified the default is 2501-2510.

/OUTPUT=*file-spec*

Specifies the output file in which to send the version information. The default is `SYSS$OUTPUT`.

Examples

The following command displays MDMS version information on the current node:

```
$ MDMS SHOW VERSION
```

The following command displays MDMS version information for nodes `COOKIE` and `DSORDS`:

```
$ MDMS SHOW VERSION /NODE=(COOKIE::, DSORDS)
```

The following command displays the version of the servers on DECnet node `SENILE`, DECnet-Plus node `PARANOID`, and TCP/IP node `DISORDERS`:

```
$ MDMS SHOW VERSION -  
_ $ /NODES=(SENILE,CORP:.DOM.PARANOID, -  
_ $ DISORDERS.COM.CORP:3001-3010)
```

MDMS SHOW VOLUME

MDMS SHOW VOLUME — Displays information about specified volumes, selected volumes or all volumes defined in the MDMS database.

Format

```
MDMS SHOW VOLUME [volume-id[,...]]
```

Equivalent **STORAGE** Command: **STORAGE SHOW VOLUME**

Parameters

volume-id

Specifies the identifiers of the volumes. All volumes are displayed if the *volume-id* parameter is omitted, subject to qualifier selection. Wildcard characters are allowed in the *volume-id*, where `*` represents a variable-length wildcard, and `%` represents a single character wildcard.

Alternatively, a volume range (separated by a dash) may be specified. A volume range is a numeric range for up to the last five characters of the volume ID. Example ranges are (ABC001-ABC250) and (T20120-T20125).

The maximum length of the volume ID is 6 characters. Valid characters are " ,! ,% -? ,A-Z ,_ ".

A comma-separated list of volume IDs or ranges may be entered.

Description

The **MDMS SHOW VOLUME** command displays information about specified volumes, selected volumes, or all volumes defined in the MDMS database.

If the *volume-id* parameter is supplied, only information on the specified volumes is displayed, which are subject to further selection based on the qualifiers.

If no *volume-id* parameter is specified, the qualifiers are used for selection. If no qualifiers are specified, all volumes are displayed.

Privileges

The request requires `MDMS_SHOW_ALL`, `MDMS_SHOW_VOLUME`, `MDMS_SHOW_POOL`, or `MDMS_SHOW_OWN`.

If the user has only MDMS_SHOW_OWN, only volumes allocated and owned by the user are displayed. If the user has MDMS_SHOW_POOL, then volumes in pools authorized to the user are displayed. To display any other volumes requires MDMS_SHOW_VOLUME or MDMS_SHOW_ALL.

Restrictions

The **/ALLOCATED**, **/NOALLOCATED**, and **/SYMBOLS** qualifiers are mutually exclusive.

The MDMS server will refuse this command if mutually exclusive qualifiers are used in the affirmative forms.

Qualifiers

/ABS_VOLSET

In a wildcard specification, this qualifier displays the ABS volume set records (those beginning with "&+") in addition to normal volumes records. By default, these pseudo-volume records are not displayed. When used with a volume name, range, or list, the associated ABS volume set records are displayed instead of the normal volume records.

/ALLOCATED **/NOALLOCATED**

Displays only volumes that are currently either allocated or not allocated respectively.

/BRIEF

Display volume information in the brief 1-line format. This is the default if no *volume-id* parameter is specified.

/FULL

Display volume information in the full multi-line format. All attributes and status fields are displayed. This is the default when a *volume-id* parameter is specified.

/OUTPUT=*file-spec*

Specifies the output file in which to send the volume information. The default is SYS\$OUTPUT.

/SCHEDULE[=*keyword*]

The **/SCHEDULE** qualifier selects volumes whose off-site or on-site date has "expired" and the volumes are not in the new location. If both dates have expired, the later of the two dates is used. The optional keyword on the schedule qualifier may be OFFSITE or ONSITE.

If the OFFSITE keyword is used, then only those volumes schedule to be moved off site are selected. If the ONSITE keyword is used, then only those volumes scheduled to be moved on site are selected. If the keyword is omitted, the volumes scheduled to be moved on site and off site are selected. Do not specify a volume ID with this qualifier.

/SYMBOLS[=*prefix*]

Stores selected selection information in process symbols. If prefix is not specified the default prefix is MDMS_INQ. The maximum length of the prefix is 8 characters. This qualifier is supported for wildcard show requests. By default there is no output on the command. If you wish output with

the symbols, use the **/OUTPUT** qualifier. The following symbols are defined on a show of a single volume:

MDMS_INQ_ACCOUNT	MDMS_INQ_ALLOCATED_DATE
MDMS_INQ_AVAILABLE_STATE	MDMS_INQ_BLOCK_FACTOR
MDMS_INQ_BRAND	MDMS_INQ_CLEANED_DATE
MDMS_INQ_CREATION_DATE	MDMS_INQ_DEALLOCATION_DATE
MDMS_INQ_DESCRIPTION	MDMS_INQ_DRIVE
MDMS_INQ_FORMAT	MDMS_INQ_FREED_DATE
MDMS_INQ_INITIALIZED_DATE	MDMS_INQ_IO_ERROR_COUNT
MDMS_INQ_JOB_NAME	MDMS_INQ_JUKEBOX_NAME
MDMS_INQ_LAST_ACCESS_DATE	MDMS_INQ_MAGAZINE_NAME
MDMS_INQ_MEDIA_TYPE	MDMS_INQ_MOUNT_COUNT
MDMS_INQ_NEXT_VOLUME	MDMS_INQ_OFFSITE_DATE
MDMS_INQ_OFFSITE_LOCATION	MDMS_INQ_ONSITE_DATE
MDMS_INQ_ONSITE_LOCATION	MDMS_INQ_OWNER_NAME
MDMS_INQ_OWNER_UIC	MDMS_INQ_PLACEMENT
MDMS_INQ_PLACENAME	MDMS_INQ_POOL_NAME
MDMS_INQ_PREVIOUS_VOLUME	MDMS_INQ_PROTECTION
MDMS_INQ_PURCHASED_DATE	MDMS_INQ_RECORD_LENGTH
MDMS_INQ_SCRATCH_DATE	MDMS_INQ_SLOTS
MDMS_INQ_SPACES	MDMS_INQ_STATE
MDMS_INQ_TIMES_CLEANED	MDMS_INQ_TRANSITION_TIME
MDMS_INQ_USER_NAME	MDMS_INQ_VOLUME_ID

Symbols are not available for access control.

On wildcard show requests, the above symbols will be suffixed by *_n*. A symbol *prefix_N* stores the number of objects that were selected by a command.

/USER_NAME=*username*

Selects volumes owned by this user.

/VOLSET

/NOVOLSET(Default)

This qualifier specifies that the show applies to all volumes of the volume set. By default, the show command only applies to specified volumes.

Examples

The following command stores selected information in DCL symbols for volume FRM001. No output is displayed by default:

```
$ MDMS SHOW VOLUME FRM001 /SYMBOLS
```

The following command displays a full listing of volume information for all volumes allocated to user HSM\$SERVER:

```
$ MDMS SHOW VOLUME /ALLOCATED /USER_NAME=HSM$SERVER /FULL
```

The following command displays a brief listing of volume information for the all volumes that are not allocated (those that are in the UNINITIALIZED, FREE, or TRANSITION states):

```
$ MDMS SHOW VOLUME /NOALLOCATED
```

The following command displays a full listing of the volumes matching the wildcard BDJ*:

```
$ MDMS SHOW VOLUME BDJ*/FULL
```

Start_Session

Start_Session — Starts an MDMS session.

Format

MDMS

Equivalent STORAGE Command: None.

Parameters

None.

Description

The **MDMS** command starts an MDMS session. The default session is a DCL session, from which you may enter multiple MDMS commands without the MDMS verb. Your prompt is MDMS>. If you enter the **/INTERFACE=GUI** qualifier, you will instead initiate a GUI session on the system.

Privileges

None.

Restrictions

None.

Qualifiers

/INTERFACE=GUI

This qualifier is required to start the Graphical User Interface on OpenVMS.

Before initiating the **MDMS/INTERFACE=GUI** command, the following command should be issued:

```
$ SET DISPLAY/CREATE/NODE=monitor-node-name/TRANSPORT=transport
```


The *monitor-node-name* parameter is the name of the node on which the monitor resides. Depending on the transport you select, you may need to enter the TCP/IP fullname, or the DECnet-Plus fullname, rather than the DECnet Phase IV node name.

The *transport* parameter is one of the following:

- DECnet – If you are using DECnet between the OpenVMS node running the GUI and the node whose monitor you are displaying the GUI.
- TCPIP – If you are using TCP/IP between the OpenVMS node running the GUI and the node whose monitor you are displaying the GUI.
- LOCAL – If the monitor is on the same node as the one running the GUI.

Examples

The following command initiates an MDMS DCL session:

```
$ MDMS
MDMS>
```

The following command initiates an MDMS GUI session on the current system, on the monitor connected to node NOD001 (which can be running an OpenVMS version supporting X-windows) running the TCP/IP protocol:

```
$ SET DISPLAY/NODE=NOD001.XYZ.COM/TRANSPORT=TCPIP
$ MDMS/INTERFACE=GUI
```

MDMS SYNCHRONIZE RESTORE

MDMS SYNCHRONIZE RESTORE — Waits for the named restore request to complete execution, then returns.

Format

```
MDMS SYNCHRONIZE RESTORE [restore-name]
```

Equivalent ABS Command: **ABS SYNCHRONIZE**

Parameters

restore-name

Specifies the name of the restore on which to synchronize.

The maximum length of the restore name is 39 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

Description

The **MDMS SYNCHRONIZE RESTORE** command waits for the named restore request to complete execution, then returns. If the restore is not currently running, the synchronize returns immediately. You can specify the **/STATUS** and **/SEVERITY** qualifiers with named symbols, which will return the transaction status and severity of the last execution of the restore request.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/SEVERITY=*symbol-name*

The **/SEVERITY** qualifier allows you to receive the severity condition of the last execution of the restore request. The severity string becomes the value of the named symbol.

/STATUS=*symbol-name*

The **/STATUS** qualifier allows you to receive the transaction status of the last execution of the restore request. The transaction status becomes the value of the named symbol.

The following are the status and severity returned by the **MDMS SYNCHRONIZE RESTORE** command:

Status	Severity
Successful completion	SUCCESS
Qualified success	WARNING
Failure status	ERROR
Abnormally terminated	FATAL

Examples

The following example starts a restore request, then waits for it to complete. On completion, the symbols MY_STATUS and MY_SEVERITY contain the transaction status and severity, respectively:

```
$ MDMS SET RESTORE NIGHTLY_RESTORE/START
$ MDMS SYNCHRONIZE RESTORE NIGHTLY_RESTORE/ -
_$ /STATUS=MY_STATUS/SEVERITY=MY_SEVERITY
$ SHOW SYMBOL MY_STATUS
MY_STATUS = "ABS_SUCCESS : Normal successful completion"
$ SHOW SYMBOL MY_SEVERITY
MY_SEVERITY = "SUCCESS"
```

MDMS SYNCHRONIZE SAVE

MDMS SYNCHRONIZE SAVE — Waits for the named save request to complete execution, then returns.

Format

MDMS SYNCHRONIZE SAVE [*save-name*]

Equivalent ABS Command: **ABS SYNCHRONIZE**

Parameters

save-name

Specifies the name of the save on which to synchronize.

Description

The **MDMS SYNCHRONIZE SAVE** command waits for the named save request to complete execution, then returns. If the save is not currently running, the synchronize returns immediately. You can specify the **/STATUS** and **/SEVERITY** qualifiers with named symbols, which will return the transaction status and severity of the last execution of the save request.

Privileges

The request requires MDMS_SHOW_ALL.

Restrictions

This command is restricted to nodes with an ABS license.

Qualifiers

/SEVERITY=*symbol-name*

The **/SEVERITY** qualifier allows you to receive the severity condition of the last execution of the save request. The severity string becomes the value of the named symbol.

/STATUS=*symbol-name*

The **/STATUS** qualifier allows you to receive the transaction status of the last execution of the save request. The transaction status becomes the value of the named symbol.

The following are the status and severity returned by the **MDMS SYNCHRONIZE SAVE** command.

Status	Severity
Successful completion	SUCCESS
Qualified success	WARNING
Failure status	ERROR
Abnormally terminated	FATAL

Examples

The following example starts a save request, then waits for it to complete. On completion, the symbols MY_STATUS and MY_SEVERITY contain the transaction status and severity, respectively:

```
$ MDMS SET SAVE NIGHTLY_SAVE/START
```

```
$ MDMS SYNCHRONIZE SAVE NIGHTLY_SAVE/ -  
_$_ /STATUS=MY_STATUS/SEVERITY=MY_SEVERITY  
$ SHOW SYMBOL MY_STATUS  
MY_STATUS = "ABS_SUCCESS : Normal successful completion"  
$ SHOW SYMBOL MY_SEVERITY  
MY_SEVERITY = "SUCCESS"
```

MDMS UNBIND VOLUME

MDMS UNBIND VOLUME — Unbinds a volume from a volume set.

Format

```
MDMS UNBIND VOLUME [volume-id]
```

Equivalent STORAGE Command: **STORAGE SPLIT**

Parameters

volume-id

Specifies the volume ID of the volume to unbind.

The maximum length of a volume ID is 6 characters. Valid characters are " ,! ,%,?-?,A-Z, _".

Description

The **MDMS UNBIND VOLUME** command unbinds a volume from a volume set. By default, when a volume is unbound from a volume set, all volumes in the volume set are unbound. To cause the volume set to be split into two volume sets, use the **/NOVOLSET** qualifier. Be aware that unbinding a volume from a volume set could cause data loss if the volume set contains valid data on every volume. Unbound volumes will remain in the **ALLOCATED** state.

Privileges

The request requires **MDMS_UNBIND_ALL** or **MDMS_UNBIND_OWN**.

If the user has only **MDMS_UNBIND_OWN**, they can unbind only those volumes allocated to them. Unbinding any other volume requires **MDMS_UNBIND_ALL**.

Restrictions

None.

Qualifiers

/USER_NAME=*username*

The user that owns the volume set. This qualifier is used to unbind a volume from a volume set on behalf of the user. The maximum length of the username is 31 characters.

This qualifier requires the right **MDMS_UNBIND_ALL**.

/VOLSET (Default)
/NOVOLSET

By default, the entire volume set containing the volume will be split into single volumes. Use the **/NOVOLSET** qualifier to split the volume set into two volume sets, with the second set beginning with the specified volume.

Note that it is not possible to unbind the first volume of a volume set using the **/NOVOLSET** qualifier.

Examples

A volume set contains volumes VOL001, VOL002, VOL003, and VOL004. The following command unbinds all the volumes in the volume set, leaving VOL001, VOL002, VOL003, and VOL004 all as single volumes. The volumes remain allocated:

```
$ MDMS UNBIND VOLUME VOL002
```

A volume set contains volumes VOL001, VOL002, VOL003, and VOL004, owned by user SMITH. The following command unbinds the volume set starting at VOL003 from the volume set. The remaining volume sets contain volumes VOL001 and VOL002 and the second set contains VOL003 and VOL004:

```
$ MDMS UNBIND VOLUME VOL003 /USER_NAME=SMITH /NOVOLSET
```

MDMS UNLOAD DRIVE

MDMS UNLOAD DRIVE — Unloads the volume contained in the specified drive.

Format

```
MDMS UNLOAD DRIVE [drive-name]
```

Equivalent STORAGE Command: **STORAGE UNLOAD DRIVE**

Parameters

drive-name

Specifies the name of the drive to unload. Specify a drive name or a logical name.

The maximum length of the drive name or logical name is 31 characters. Valid characters are "\$,-,0-9,A-Z,_,a-z".

Description

The **MDMS UNLOAD DRIVE** command unloads the volume currently contained in the specified drive.

Privileges

The request requires MDMS_UNLOAD_ALL.

MDMS_ASSIST is also required unless the **/NOASSIST** qualifier is specified.

Restrictions

None.

Qualifiers

/ASSIST (Default)

/NOASSIST

The default **/ASSIST** qualifier requests operator assistance to prompt the operator to unload the volume from the drive. If **/NOASSIST** is specified, the operator is not notified. This qualifier requires the right MDMS_ASSIST.

/REPLY=*symbol*

The name of the symbol to receive the operator's reply when operator intervention is needed. The symbol will contain the operator reply to the DCL **REPLY/TO** or **REPLY/ABORT** command. The maximum length of a symbol name is 31 characters.

This qualifier is only applicable when the **/ASSIST** qualifier is specified.

/WAIT (Default)

/NOWAIT

The **/NOWAIT** qualifier returns an informational message indicating that the unload is being queued. The **/WAIT** qualifier causes the **MDMS UNLOAD DRIVE** command to wait until a drive is unloaded.

Examples

The following command unloads the volume in drive **\$1\$MUA151:**. There is no timeout value, and no operator assistance is requested:

```
$ MDMS UNLOAD DRIVE $1$MUA151:
```

MDMS UNLOAD VOLUME

MDMS UNLOAD VOLUME — Unloads the specified volume from a drive.

Format

```
MDMS UNLOAD VOLUME [volume-id]
```

Equivalent STORAGE Command: **STORAGE UNLOAD VOLUME**

Parameters

volume-id

Specifies the volume ID of the volume to unload.

The maximum length of the volume ID is 6 characters. Valid characters are " , ! , " , % - ? , A - Z , _ " .

Note

If there is a discrepancy between the Load Volume and Volume fields in the display of a **SHOW DRIVE** command, attempt to unload the Load Volume field first.

If this is unsuccessful, attempt to unload the Volume field next.

If the issue persists, execute an **UNLOAD DRIVE** command.

Description

The **MDMS UNLOAD VOLUME** command will unload a volume from a drive. If the volume resides in a jukebox, it will be returned to its jukebox slot.

Privileges

The request requires MDMS_UNLOAD_ALL, MDMS_UNLOAD_POOL, or MDMS_UNLOAD_OWN.

If the user only has MDMS_UNLOAD_OWN, only those volumes allocated to the user can be unloaded. With MDMS_UNLOAD_POOL, the user can unload a volume in a pool to which they are authorized. To unload any other volume requires MDMS_UNLOAD_ALL.

MDMS_ASSIST is also required unless the **/NOASSIST** qualifier is specified.

Restrictions

The volume cannot be unloaded if it is currently mounted.

Qualifiers

/ASSIST (Default)
/NOASSIST

The default **/ASSIST** qualifier requests operator assistance to prompt the operator to unload the volume from the drive. If **/NOASSIST** is specified, the operator is not notified.

This qualifier requires the right MDMS_ASSIST.

/REPLY=*symbol*

The name of the symbol to receive the operator's reply when operator intervention is needed. The symbol will contain the operator reply to the DCL **REPLY/TO** or **REPLY/ABORT** command. The maximum length of a symbol name is 31 characters.

This qualifier is only applicable when the **/ASSIST** qualifier is specified.

/WAIT (Default)
/NOWAIT

The **/NOWAIT** qualifier returns an informational message indicating that the unload is being queued. The **/WAIT** qualifier causes the **MDMS UNLOAD VOLUME** command to wait until a volume is unloaded.

Examples

The following command unloads volume VOL003 from the drive on which it is loaded and the command does not wait until the unload is complete:

```
$ MDMS UNLOAD VOLUME VOL003 /NOWAIT
```


Chapter 2. ABS to MDMS Command Mappings

This chapter shows the mappings from the ABS DCL commands to the MDMS DCL commands.

Important

VSI recommends using MDMS commands in preference to ABS commands whenever possible. MDMS provides a consistent and integrated interface for system management. ABS commands are retained for compatibility and may provide more limited functionality.

2.1. ABS CREATE CATALOG

Table 2.1. ABS CREATE CATALOG

RUN SYS\$SYSTEM:ABS\$CATALOG_OBJECT		MDMS CREATE CATALOG	
Function Name: CREATE		N/A	
Catalog Name: NAME		NAME (Parameter)	
Catalog Type: TYPE		/TYPE=	
	BRIEF		FILES
	FULL_RESTORE		DISKS
	ORACLE_DB		ORACLE_DB
	SLS		SLS
	VOLUME_SET		VOLUME_SETS
Catalog Owner: OWNER		/[NO]OWNER_NAME	
Use faster staging catalog operation:		/[NO]STAGING	
Catalog Location: DIRECTORY		/DIRECTORY	
N/A		/[NO]ACCESS_CONTROL	
N/A			USERNAME
N/A			ALLOW
N/A		/NODE	

2.2. ABS CREATE ENVIRONMENT

Table 2.2. ABS CREATE ENVIRONMENT

ABS CREATE ENVIRONMENT		MDMS CREATE ENVIRONMENT	
NAME (Parameter)		NAME (Parameter)	
/ACCESS_CONTROL		/[NO]ACCESS_CONTROL	
	USER_ID		USERNAME

ABS CREATE ENVIRONMENT		MDMS CREATE ENVIRONMENT	
	ACCESS		ALLOW
/ACTION		/ACTION	
/COMPRESSION		/COMPRESSION	
/DATA_SAFETY		/[NO]DATA_SAFETY	
/DRIVE_COUNT		/DRIVE_COUNT	
/[NO]EPILOGUE		/[NO]EPILOGUE	
/INTERVAL_FOR_RETRY		/INTERVAL	
/[NO]LINKS_ONLY		/[NO]LINKS_ONLY	
/LISTING_OPTION		/LISTING_OPTION	
/[NO]LOCK_OPTION		/[NO]LOCK	
/[NO]NOTIFICATION		/[NO]NOTIFICATION	
	METHOD		N/A
	RECEIVER		MAIL/OPCOM
	TYPE		TYPE
	WHEN		WHEN
/OWNER		/[NO]OWNER_NAME	
/PROFILE		/PROFILE	
	CLUSTER		CLUSTER
	NODE		NODE
	PRIVS		PRIVILEGES
	RIGHTS_LIST		RIGHTS_LIST
	USERNAME		USER
/[NO]PROLOGUE		/[NO]PROLOGUE	
/RETRY_COUNT		/RETRY_COUNT	
/SPAN_FILESYSTEMS		/SPAN_FILESYSTEMS	
N/A		/[NO]ASSIST	
N/A		/DESCRIPTION	
N/A		/INHERIT	

2.3. ABS CREATE STORAGE_CLASS

Table 2.3. ABS CREATE STORAGE_CLASS

ABS CREATE STORAGE_CLASS	MDMS CREATE ARCHIVE
NAME (Parameter)	NAME (Parameter)
/ACCESS_CONTROL	/[NO]ACCESS_CONTROL

ABS CREATE STORAGE_CLASS		MDMS CREATE ARCHIVE	
	USER_ID		USERNAME
	ACCESS		ALLOW
/ARCHIVE_TYPE		/ARCHIVE_TYPE	
/CATALOG		/CATALOG	
/CONSOLIDATION		/CONSOLIDATION	
	INTERVAL		INTERVAL
	COUNT		SAVESETS
	SIZE		VOLUMES
/[NO]DRIVE_LIST		/[NO]DRIVES	
/[NO]EXPIRATION		/[NO]EXPIRATION_DATE	
/LOCATION		/DESTINATION	
/MAXIMUM_SAVES		/MAXIMUM_SAVES	
/[NO]MEDIA_LOCATION		/[NO]LOCATION	
/OWNER		/[NO]OWNER_NAME	
/RETAIN		/RETENTION_DAYS	
/[NO]TAPE_POOL		/[NO]POOL	
/[NO]TYPE_OF_MEDIA		/MEDIA_TYPE	
/[NO]VOLUME_SET		/[NO]VOLUME_SETS	
N/A		/DESCRIPTION	
N/A		/INHERIT	

2.4. ABS DELETE

Table 2.4. ABS DELETE

ABS DELETE Object	MDMS DELETE Object
NAME (Parameter)	NAME (Parameter)
/CONFIRM	N/A
/OVERRIDE_REFERENCE_COUNT	/OVERRIDE
/VERSION	N/A
N/A	/SCHEDULE (Save/Restore)

2.5. ABS DELETE CATALOG

Table 2.5. ABS DELETE CATALOG

RUN SYSS\$SYSTEM:ABS\$CATALOG_OBJECT	MDMS DELETE CATALOG
Function Name: DELETE	N/A

RUN SYS\$SYSTEM:ABS\$CATALOG_OBJECT	MDMS DELETE CATALOG
Catalog Name: NAME	NAME (Parameter)
Delete Catalog NAME 9YES, [NO]0:	N/A
N/A	/NODE
N/A	/OVERRIDE

2.6. ABS LOOKUP

Table 2.6. ABS LOOKUP

ABS LOOKUP	MDMS SHOW CATALOG
DATA_OBJECT (Parameter)	/FILES
/ALL	/ALL
/BRIEF	/BRIEF
/FULL	/FULL
/[NO] OUTPUT	/OUTPUT
/[NO] SUMMARY	N/A
/BEFORE (Positional)	/BEFORE
/CATALOG (Positional)	NAME (Parameter)
/DATE_ARCHIVED (Positional)	N/A
/EXPIRATION (Positional)	N/A
/OBJECT_TYPE (Positional)	/DATA_SELECT_TYPE
/NODE (Positional)	/SOURCE_NODE
/SINCE (Positional)	/SINCE
/STORAGE_CLASS (Positional)	/ARCHIVE
N/A	/NODES
N/A	/SYMBOLS

2.7. ABS REPORT SAVE_LOG

Table 2.7. ABS REPORT SAVE_LOG

ABS REPORT SAVE_LOG	MDMS SHOW CATALOG
/BEFORE	/BEFORE
/BRIEF	/BRIEF
/CATALOG	NAME (Parameter)
/DATE_ARCHIVED	N/A
/FULL	/FULL

ABS REPORT SAVE_LOG	MDMS SHOW CATALOG
/INCLUDE	/INCLUDE
/[NO]OUTPUT	/OUTPUT
/REQUEST	/SAVES
/SINCE	/SINCE
/STORAGE_CLASS	/ARCHIVE
/[NO]SUMMARY	N/A
/VOLUME	/VOLUMES
N/A	/NODES
N/A	/SYMBOLS

2.8. ABS RESTORE

Table 2.8. ABS RESTORE

ABS RESTORE		MDMS CREATE RESTORE	
INCLUDE_SPEC (Parameter)		/[NO]INCLUDE ¹	
/ACCESS_CONTROL		/[NO]ACCESS_CONTROL	
	USER_ID		USERNAME
	ACCESS		ALLOW
/[NO]BEFORE		/[NO]BEFORE_DATE	
/CATALOG		/[NO]CATALOG	
/DATE_ARCHIVED		/[NO]DATE_ARCHIVED	
/DESTINATION		/DESTINATION	
/ENVIRONMENT		/ENVIRONMENT	
/[NO]EPILOGUE		/[NO]EPILOGUE	
/EXECUTION_NODE		/[NO]EXECUTION_NODES	
/FULL		N/A	
/INCREMENTAL		/[NO]INCREMENTAL	
/LATEST_COPY		N/A	
/NAME		NAME (Parameter)	
/OWNER		/[NO]OWNER_NAME	
/[NO]PROLOGUE		/[NO]PROLOGUE	
/SELECTIVE		N/A	
/SEQUENCE_OPTION		/SEQUENCE_OPTION	
/[NO]SINCE		/[NO]SINCE_DATE	
/[NO]START_TIME		/[NO]START_DATE	

ABS RESTORE	MDMS CREATE RESTORE
/STORAGE_CLASS	/ARCHIVE
/AGENT_QUALIFIERS (Positional)	N/A (Selection Object)
/CONFLICT_OPTIONS (Positional)	N/A (Selection Object)
/OBJECT_TYPE (Positional)	/DATA_SELECT_TYPE ¹
/TARGET_NODE (Positional)	/[NO]SOURCE_NODE ¹
N/A	/[NO]BASE_DATE
N/A	/DELETE_INTERVAL
N/A	/DESCRIPTION
N/A	/[NO]EXCLUDE ¹
N/A	/[NO]EXPLICIT_INTERVAL
N/A	/FREQUENCY
N/A	/[NO]GROUPS
N/A	/INHERIT
N/A	/KEEP
N/A	/[NO]SELECTIONS
N/A	/[NO]SKIP_TIME

¹Information is stored in the default selection.

2.9. ABS SAVE

Table 2.9. ABS SAVE

ABS SAVE		MDMS CREATE SAVE	
NAME (Parameter)		NAME (Parameter)	
INCLUDE_SPEC (Parameter)		/[NO]INCLUDE ¹	
/ACCESS_CONTROL		/[NO]ACCESS_CONTROL	
	USER_ID		USERNAME
	ACCESS		ALLOW
/[NO]CONFIRM		N/A	
/ENVIRONMENT		/ENVIRONMENT	
/[NO]EPILOGUE		/[NO]EPILOGUE	
/EXECUTION_NODE		/[NO]EXECUTION_NODES	
/EXPLICIT		/[NO]EXPLICIT_INTERVAL	
/FULL		N/A	
/INCREMENTAL		/[NO]INCREMENTAL	
/OWNER		/[NO]OWNER_NAME	

ABS SAVE	MDMS CREATE SAVE
/[NO]PROLOGUE	/[NO]PROLOGUE
/SCHEDULE_OPTION	/FREQUENCY
/SELECTIVE	N/A
/SEQUENCE_OPTION	/SEQUENCE_OPTION
/[NO]START_TIME	/[NO]START_DATE
/STORAGE_CLASS	/ARCHIVE
/ADD (include_spec)	/ADD
/AGENT_QUALIFIERS (Positional)	N/A (Selection Object)
/[NO]BEFORE (Positional)	N/A (Selection Object)
/EXCLUDE (Positional)	/[NO]EXCLUDE ¹
/OBJECT_TYPE (Positional)	/DATA_SELECT_TYPE ¹
/REMOVE (access_control & include_spec)	/REMOVE
/[NO]SINCE (Positional)	N/A (Selection Object)
/SOURCE_NODE (Positional)	/[NO]SOURCE_NODE ¹
N/A	/[NO]BASE_DATE
N/A	/DELETE_INTERVAL
N/A	/DESCRIPTION
N/A	/[NO]GROUPS
N/A	/KEEP
N/A	/REPLACE
N/A	/[NO]SELECTIONS
N/A	/[NO]SKIP_TIME

¹Information is stored in the default selection.

2.10. ABS SET CATALOG

Table 2.10. ABS SET CATALOG

RUN SYS\$SYSTEM:ABS\$CATALOG_OBJECT	MDMS SET CATALOG
Function Name: MODIFY	N/A
Catalog Name: NAME	NAME (Parameter)
Catalog Owner: OWNER	/[NO]OWNER_NAME
Use faster staging catalog operation:	/[NO]STAGING
N/A	/[NO]ACCESS_CONTROL
N/A	USERNAME
N/A	ALLOW

RUN SYS\$SYSTEM:ABS\$CATALOG_OBJECT	MDMS SET CATALOG
N/A	/NODE

2.11. ABS SET ENVIRONMENT

Table 2.11. ABS SET ENVIRONMENT

ABS SET ENVIRONMENT		MDMS SET ENVIRONMENT	
NAME (Parameter)		NAME (Parameter)	
/ACCESS_CONTROL		/[NO]ACCESS_CONTROL	
	USER_ID		USERNAME
	ACCESS		ALLOW
/ACTION		/ACTION	
/COMPRESSION		/COMPRESSION	
/[NO]CONFIRM		N/A	
/DATA_SAFETY		/[NO]DATA_SAFETY	
/DRIVE_COUNT		/DRIVE_COUNT	
/[NO]EPILOGUE		/[NO]EPILOGUE	
/INTERVAL_FOR_RETRY		/INTERVAL	
/[NO]LINKS_ONLY		/[NO]LINKS_ONLY	
/LISTING_OPTION		/LISTING_OPTION	
/[NO]LOCK_OPTION		/[NO]LOCK	
/[NO]NOTIFICATION		/[NO]NOTIFICATION	
	METHOD		N/A
	RECEIVER		MAIL/OPCOM
	TYPE		TYPE
	WHEN		WHEN
/OWNER		/[NO]OWNER_NAME	
/PROFILE		/PROFILE	
	CLUSTER		CLUSTER
	NODE		NODE
	PRIVS		PRIVILEGES
	RIGHTS_LIST		RIGHTS_LIST
	USERNAME		USER
/[NO]PROLOGUE		/[NO]PROLOGUE	
/REMOVE(access_control)		/REMOVE	
/RETRY_COUNT		/RETRY_COUNT	

ABS SET ENVIRONMENT	MDMS SET ENVIRONMENT
/SPAN_FILESYSTEMS	/SPAN_FILESYSTEMS
N/A	/ADD
N/A	/[NO]ASSIST
N/A	/DESCRIPTION
N/A	/REPLACE

2.12. ABS SET RESTORE

Table 2.12. ABS SET RESTORE

ABS SET RESTORE		MDMS SET RESTORE	
NAME (Parameter)		NAME (Parameter)	
INCLUDE_SPEC (Parameter)		/[NO]INCLUDE ¹	
/ACCESS_CONTROL		/[NO]ACCESS_CONTROL	
	USER_ID		USERNAME
	ACCESS		ALLOW
/[NO]BEFORE		/[NO]BEFORE_DATE	
/CATALOG		/[NO]CATALOG	
/[NO]CONFIRM		N/A	
/DATE_ARCHIVED		/[NO]DATE_ARCHIVED	
/DESTINATION		/DESTINATION	
/ENVIRONMENT		/ENVIRONMENT	
/[NO]EPILOGUE		/[NO]EPILOGUE	
/EXECUTION_NODE		/[NO]EXECUTION_NODES	
/FULL		N/A	
/INCREMENTAL		/[NO]INCREMENTAL	
/LATEST_COPY		N/A	
/OWNER		/[NO]OWNER_NAME	
/[NO] PROLOGUE		/[NO]PROLOGUE	
/SELECTIVE		N/A	
/SEQUENCE_OPTION		/SEQUENCE_OPTION	
/[NO]SINCE		/[NO]SINCE_DATE	
/[NO]START_TIME		/[NO]START_DATE	
/STORAGE_CLASS		/ARCHIVE	
/ADD (include_spec)		/ADD	
/AGENT_QUALIFIERS (Positional)		N/A (Selection Object)	

ABS SET RESTORE	MDMS SET RESTORE
/CONFLICT_OPTIONS (Positional)	N/A (Selection Object)
/OBJECT_TYPE (Positional)	/DATA_SELECT_TYPE ¹
/REMOVE (access_control & include_spec)	/REMOVE
/TARGET_NODE (Positional)	/[NO]SOURCE_NODE ¹
N/A	/[NO]BASE_DATE
N/A	/DELETE_INTERVAL
N/A	/DESCRIPTION
N/A	/[NO]EXCLUDE ¹
N/A	/[NO]EXPLICIT_INTERVAL
N/A	/FREQUENCY
N/A	/[NO]GROUPS
N/A	/KEEP
N/A	/REPLACE
N/A	/[NO]SELECTIONS
N/A	/[NO]SKIP_TIME

¹Information is stored in the default selection.

2.13. ABS SET SAVE

Table 2.13. ABS SET SAVE

ABS SET SAVE		MDMS SET SAVE	
INCLUDE_SPEC (Parameter)		/[NO]INCLUDE ¹	
/ACCESS_CONTROL		/[NO]ACCESS_CONTROL	
	USER_ID		USERNAME
	ACCESS		ALLOW
/ENVIRONMENT		/ENVIRONMENT	
/[NO]EPILOGUE		/[NO]EPILOGUE	
/EXECUTION_NODE		/[NO]EXECUTION_NODES	
/EXPLICIT		/[NO]EXPLICIT_INTERVAL	
/FULL		N/A	
/INCREMENTAL		/[NO]INCREMENTAL	
/NAME		NAME (Parameter)	
/OWNER		/[NO]OWNER_NAME	
/[NO]PROLOGUE		/[NO]PROLOGUE	
/SCHEDULE_OPTION		/FREQUENCY	

ABS SET SAVE	MDMS SET SAVE
/SELECTIVE	N/A
/SEQUENCE_OPTION	/SEQUENCE_OPTION
/[NO]START_TIME	/[NO]START_DATE
/STORAGE_CLASS	/ARCHIVE
/AGENT_QUALIFIERS (Positional)	N/A (Selection Object)
/[NO]BEFORE (Positional)	N/A (Selection Object)
/EXCLUDE (Positional)	/[NO]EXCLUDE ¹
/OBJECT_TYPE (Positional)	/DATA_SELECT_TYPE ¹
/[NO]SINCE (Positional)	N/A (Selection Object)
/SOURCE_NODE (Positional)	/[NO]SOURCE_NODE ¹
N/A	/[NO]BASE_DATE
N/A	/DELETE_INTERVAL
N/A	/DESCRIPTION
N/A	/[NO]GROUPS
N/A	/INHERIT
N/A	/KEEP
N/A	/[NO]SELECTIONS
N/A	/[NO]SKIP_TIME

¹Information is stored in the default selection.

2.14. ABS SET STORAGE_CLASS

Table 2.14. ABS SET STORAGE_CLASS

ABS SET STORAGE_CLASS		MDMS SET ARCHIVE	
NAME (Parameter)		NAME (Parameter)	
/ACCESS_CONTROL		/[NO]ACCESS_CONTROL	
	USER_ID		USERNAME
	ACCESS		ALLOW
/ARCHIVE_TYPE		/ARCHIVE_TYPE	
/CATALOG		/CATALOG	
/[NO]CONFIRM		N/A	
/CONSOLIDATION		/CONSOLIDATION	
	INTERVAL		INTERVAL
	COUNT		SAVESETS
	SIZE		VOLUMES

ABS SET STORAGE_CLASS	MDMS SET ARCHIVE
/[NO]DRIVE_LIST	/[NO]DRIVES
/[NO]EXPIRATION	/[NO]EXPIRATION_DATE
/LOCATION	/DESTINATION
/MAXIMUM_SAVES	/MAXIMUM_SAVES
/[NO]MEDIA_LOCATION	/[NO]LOCATION
/OWNER	/[NO]OWNER_NAME
/REMOVE(access_control)	/REMOVE
/RETAIN	/RETENTION_DAYS
/[NO]TAPE_POOL	/[NO]POOL
/[NO]TYPE_OF_MEDIA	/MEDIA_TYPE
/[NO]VOLUME_SET	/[NO]VOLUME_SETS
N/A	/ADD
N/A	/DESCRIPTION
N/A	/REPLACE

2.15. ABS SHOW CATALOG

Table 2.15. ABS SHOW CATALOG

RUN SYS\$SYSTEM:ABS\$CATALOG_OBJECT	MDMS SHOW CATALOG
Function Name: SHOW	N/A
Catalog Name: NAME	NAME (Parameter)
N/A	/BRIEF
N/A	/FULL
N/A	/NODE
N/A	/OUTPUT
N/A	/SYMBOLS

2.16. ABS SHOW ENVIRONMENT

Table 2.16. ABS SHOW ENVIRONMENT

ABS SHOW ENVIRONMENT	MDMS SHOW ENVIRONMENT
NAME (Parameter)	NAME (Parameter)
/BRIEF	/BRIEF
/FULL	/FULL
/OUTPUT	/OUTPUT

ABS SHOW ENVIRONMENT	MDMS SHOW ENVIRONMENT
/VERSION	N/A
N/A	/SYMBOLS

2.17. ABS SHOW REQUEST_STATUS

Table 2.17. ABS SHOW REQUEST_STATUS

ABS SHOW REQUEST_STATUS	MDMS SHOW REQUEST
REQUEST_NAME (Parameter)	REQUEST_ID (Parameter)

2.18. ABS SHOW RESTORE

Table 2.18. ABS SHOW RESTORE

ABS SHOW RESTORE	MDMS SHOW RESTORE
NAME (Parameter)	NAME (Parameter)
/BRIEF	BRIEF
/FULL	/FULL
/OUTPUT	/OUTPUT
/PROFILE	/PROFILE
/SYMBOLS	/SYMBOLS
/VERSION	N/A
N/A	/SELECTIONS

2.19. ABS SHOW SAVE

Table 2.19. ABS SHOW SAVE

ABS SHOW SAVE	MDMS SHOW SAVE
NAME (Parameter)	NAME (Parameter)
/BRIEF	BRIEF
/FULL	/FULL
/OUTPUT	/OUTPUT
/PROFILE	/PROFILE
/SYMBOLS	/SYMBOLS
/VERSION	N/A
N/A	/SELECTIONS

2.20. ABS SHOW STORAGE_CLASS

Table 2.20. ABS SHOW STORAGE_CLASS

ABS SHOW STORAGE_CLASS	MDMS SHOW ARCHIVE
NAME (Parameter)	NAME (Parameter)
/BRIEF	BRIEF
/FULL	/FULL
/OUTPUT	/OUTPUT
/VERSION	/N/A
N/A	/SYMBOLS

2.21. ABS SYNCHRONIZE

Table 2.21. ABS SYNCHRONIZE

ABS SYNCHRONIZE	MDMS SYNCHRONIZE RESTORE MDMS SYNCHRONIZE SAVE
NAME (Parameter)	NAME (Parameter)
/SEVERITY	/SEVERITY
/STATUS	/STATUS

Chapter 3. MDMS Rights and Privileges

This chapter explains MDMS user rights and privileges. Every MDMS user/potential user will be assigned zero or more rights in their SYSUAF file. These rights will be examined on a per-command basis to determine whether a user has sufficient privilege to issue a command. The command is accepted for processing only if the user has sufficient privilege. In cases where the user has no rights, the entire command is rejected. Each right has a name with the following format:

`MDMS_rightname.`

Rights are looked-up on the client OpenVMS node that receives the request. As such, each user must have an account on the client node.

- In the case of DCL commands and applications, this would be the node at which the request is issued.
- From the GUI, it is the node whose MDMS\$SERVER process receives the request. The rights are translated into a bitmap and passed to the database server for validation.

3.1. Types of MDMS Rights

MDMS has the following rights:

- High-level rights
- Low-level rights
- ABS rights

3.1.1. High-Level Rights

These rights are designed for a specific kind of user, to support a typical MDMS installation, and make the assignments of rights to users easy. The three high-level MDMS rights, the default right, administrator right, and the additional right are described in the following table:

Table 3.1. High-Level Rights

High-Level Right	Allows Privilege to...
MDMS_USER	A non-privileged MDMS user who wants to use MDMS to manage tape volumes for BACKUP, ABS, or HSM purposes.
MDMS_APPLICATION	Main applications that MDMS supports—ABS and HSM server processes.
MDMS_OPERATOR	The user responsible for day-to-day operations in the MDMS environment.
Default Right (A hidden high-level right)	The low-level rights contained in it, for users with no MDMS rights. They are additional to any specific rights a user may have been granted.

High-Level Right	Allows Privilege to...
	This is the default right. By default, there are no low-level rights assigned to the default right. If rights are assigned to the default right, they apply to all users in the system, since every user is effectively granted the default right. The default right can be disabled with the MDMS_NO_DEFAULT identifier in a user's UAF file.
MDMS_ALL_RIGHTS (Administrator right)	A system administrator to perform any operation. MDMS_ALL_RIGHTS can be enabled with the OpenVMS SYSPRV privilege.
Additional Right	All operations.

You can disable the mapping of SYSPRV to MDMS_ALL_RIGHTS using the **SET DOMAIN** command.

3.1.2. Low-Level Rights

Each command or command option will be tagged with one or more low-level rights that are needed to perform the operation. Where more than one right is specified, the command indicates the appropriate combination of rights needed. The MDMS administrator can assign a set of low-level rights to each high-level right. The administrator can then simply assign the high-level right to the user.

MDMS translates the high-level right to respective low-level rights while processing a command. For additional flexibility, the user can be assigned a combination of high-level and low-level rights. The result will be a sum of all rights defined.

The default set of mapping of high-level to low-level rights will be assigned at installation (by default) and stored in the domain record. However, the MDMS administrator can change these assignments by using the **SET DOMAIN** command.

Note

By default, a user has no rights and cannot use MDMS. The system administrator can change the user's rights using the **SET DOMAIN** command. These rights can again be disabled on a per-user basis as needed.

The low-level rights are designed to be applied to operations. A given command, with a given set of qualifiers or options, requires the sum of the rights needed for the command and all supplied options. In many cases, some options require more privilege than the command, and that higher privilege will be applied to the entire command if those options are specified. Available low-level rights are described in the following table:

Table 3.2. Low Level Rights

Low Level Right Name	Allows Privilege to...
MDMS_ALL_RIGHTS	Enable all operations. This right is for the system administrator.
MDMS_ALLOCATE_ALL	Allocate volumes or drives for any user.

Low Level Right Name	Allows Privilege to...
MDMS_ALLOCATE_OWN	Allocate a drive and become its "owner".
MDMS_ALLOCATE_POOL	Allocate a volume from an authorized pool.
MDMS_ASSIST	Request operator assistance on calls.
MDMS_BIND_ALL	Bind any volumes together in a volume set.
MDMS_BIND_OWN	Bind owned volumes together in a volume set.
MDMS_CANCEL_ALL	Cancel any request.
MDMS_CANCEL_OWN	Cancel one's own requests.
MDMS_CANCEL_POOL	Cancel a request of a member of the same pool.
MDMS_CREATE_ALL	Create any database object.
MDMS_CREATE_POOL	Create volumes in a pool authorized to user.
MDMS_DEALLOCATE_ALL	Deallocate volumes for any user.
MDMS_DEALLOCATE_OWN	Deallocate an owned volume or drive.
MDMS_DELETE_ALL	Delete any database object.
MDMS_DELETE_POOL	Delete volumes in pool authorized to user.
MDMS_INITIALIZE_ALL	Initialize any volume.
MDMS_INITIALIZE_POOL	Initialize a volume in pool authorized to user.
MDMS_INVENTORY_ALL	Perform inventory on any jukebox.
MDMS_LOAD_ALL	Load any volumes including scratch volumes.
MDMS_LOAD_OWN	Load owned volumes into drives.
MDMS_LOAD_POOL	Load volumes in pool authorized to user.
MDMS_LOAD_SCRATCH	Load scratch volumes.
MDMS_MOVE_ALL	Move any volume.
MDMS_MOVE_OWN	Move owned volumes.
MDMS_MOVE_POOL	Move volumes in pool authorized to user.
MDMS_SET_ALL	SET (modify) any database object.
MDMS_SET_PROTECTED	SET internal MDMS attributes in an object.
MDMS_SET_OWN	SET (modify) volumes allocated to user.
MDMS_SET_POOL	SET (modify) volumes in pool authorized to user.
MDMS_SET_RIGHTS	SET (modify) rights in the domain.
MDMS_SHOW_ALL	SHOW or REPORT any database object.
MDMS_SHOW_OWN	SHOW or REPORT volumes allocated to user.
MDMS_SHOW_POOL	SHOW or REPORT volumes in pool authorized to user.
MDMS_SHOW_RIGHTS	Show rights with the SHOW DOMAIN/FULL command.

Low Level Right Name	Allows Privilege to...
MDMS_UNBIND_ALL	Unbind any volumes.
MDMS_UNBIND_OWN	Unbind owned objects from a volume set.
MDMS_UNLOAD_ALL	Unload any volumes or drives.
MDMS_UNLOAD_OWN	Unload volumes allocated to user from a drive.
MDMS_UNLOAD_POOL	Unload volumes in pool authorized to user.

3.1.3. ABS Rights

MDMS can be defined to recognize ABS rights and map them to MDMS rights. This capability is disabled by default and can be enabled with the **SET DOMAIN** command. The exact mapping for ABS rights is described in the following table:

Table 3.3. ABS to MDMS Rights

ABS Rights	MDMS Rights
ABS_BACKUP_JOB	MDMS_USER
ABS_BYPASS	MDMS_ALL_RIGHTS
ABS_CREATE_EXECUTION_ENV	MDMS_CREATE_ALL MDMS_SET_ALL MDMS_SHOW_ALL
ABS_CREATE_REMOTE_JOB	None.
ABS_CREATE_STORAGE_CLASS	MDMS_CREATE_ALL MDMS_SET_ALL MDMS_SHOW_ALL
ABS_LOOKUP_ALL	None.
ABS_SHOW_ALL	MDMS_SHOW_ALL

3.2. Default High-Level to Low-Level Mapping

This section defines the default high- to low-level mapping for each high-level right.

3.2.1. MDMS_USER Rights

Table 3.4. MDMS_USER Rights

MDMS User...	Allows Privilege to...
MDMS_ALLOCATE_OWN	Allocate a drive and become the "owner".
MDMS_ALLOCATE_POOL	Allocate a volume from a pool authorized to user.
MDMS_ASSIST	Request operator assistance on calls.

MDMS User...	Allows Privilege to...
MDMS_BIND_OWN	Bind owned volumes together in a volume set.
MDMS_CANCEL_OWN	Cancel one's own requests.
MDMS_DEALLOCATE_OWN	Deallocate an owned volume or drive.
MDMS_LOAD_OWN	Load owned volumes into drives.
MDMS_SHOW_OWN	SHOW or REPORT volumes allocated to user.
MDMS_SHOW_POOL	SHOW or REPORT volumes in pool authorized to user.
MDMS_UNBIND_OWN	Unbind owned objects from a volume set.
MDMS_UNLOAD_OWN	Unload volumes allocated to user from a drive.

3.2.2. MDMS_OPERATOR Rights

Table 3.5. MDMS_OPERATOR Rights

MDMS Operator...	Allows Privilege to...
MDMS_ALLOCATE_ALL	Allocate volumes or drives for any user.
MDMS_ASSIST	Request operator assistance on calls.
MDMS_BIND_ALL	Bind any volumes together in a volume set.
MDMS_CANCEL_ALL	Cancel any request.
MDMS_CREATE_POOL	Create volumes in a pool authorized to user.
MDMS_DEALLOCATE_ALL	Deallocate volumes for any user.
MDMS_DELETE_POOL	Delete volumes in pool authorized to user.
MDMS_INITIALIZE_ALL	Initialize any volume.
MDMS_INVENTORY_ALL	Perform inventory on any jukebox.
MDMS_LOAD_ALL	Load any volumes including scratch volumes.
MDMS_MOVE_ALL	Move any volume.
MDMS_SET_OWN	SET (modify) volumes allocated to user.
MDMS_SET_POOL	SET (modify) volumes in pool authorized to user.
MDMS_SHOW_ALL	SHOW or REPORT any database object.
MDMS_SHOW_RIGHTS	Show rights with the SHOW DOMAIN/FULL command.
MDMS_UNBIND_ALL	Unbind any volumes.
MDMS_UNLOAD_ALL	Unload any volumes or drives.

3.2.2.1. Domain Commands for Mapping Privileges

```
SET DOMAIN
    /NO]ABS_RIGHTS
    /ADD
```

```
[NO]APPLICATION_RIGHTS [= (right[, ...]) ]  
[NO]DEFAULT_RIGHTS [= (right[, ...]) ]  
[NO]OPERATOR_RIGHTS [= (right[, ...]) ]  
/REMOVE  
[NO]SYSPRV  
[NO]USER_RIGHTS [= (right[, ...]) ]
```

The following command adds the MDMS_SET_PROTECTED right to the operator rights list:

```
$ SET DOMAIN /OPERATOR_RIGHTS=MDMS_SET_PROTECTED /ADD
```

Chapter 4. MDMS Logical Names

Logical Name	Definition and Description
MDMS\$DATABASE_LOCATION MDMS\$DB	These logical names point to the location of the MDMS database files.
MDMS\$DATABASE_SERVERS	This logical name is a comma-separated list of full node names of potential database servers. When a server starts up, it uses this logical to see if it may be a database server. If the server finds its node name in the list, it tries to become the database server. If the server does not find itself in the list, it then knows that it is not a database server, and will then tries to communicate with the node in the list to find the database server. The name of the node defines how the two server communicate with each other. This list of names must be DECnet Phase IV, DECnet-Plus, or TCP/IP node names. They can be a mix of different protocols or the same. For example, the node list could look as follows: <pre>NODE1,NODE2.SITE.INC.COM, - INC:.SITE.NODE3</pre> The above example shows that to communicate with: ● NODE1 – Use DECnet Phase IV ● NODE2 – Use DECnet-Plus ● NODE3 – Use TCP/IP Following are examples of valid DECnet and TCPIP names. Valid DECnet node names: <pre>DEC:.CXO.FARMS[::] - DECnet-Plus NABSCO[::] - DECnet Phase IV</pre> Note that the DECnet node name is terminated at the ":", if present. Valid TCP/IP node names: <pre>nabsc0-12.cxo.dec.com nabsc0-12[.cxo.dec.com]: nabsc0-12[.cxo.dec.com]:2501 nabsc0-12[.cxo.dec.com]:2501-2510</pre>

Logical Name	Definition and Description
MDMS\$LOGFILE_LOCATION MDMS\$LOG	These logical names point to the location of the MDMS log files.
MDMS\$LOGFILTER	A bitmask that defines the level of logging to be performed by the MDMS\$SERVER process. If not defined, the default value is 15, which turns on completion audit logging of requests that change the database, and event and serious error logging. See MDMS\$SYSTARTUP.COM for bitmask definitions.
MDMS\$MAILBOX	This logical name is the name of the mailbox used by the MDMS\$SERVER process.
MDMS\$ROOT	This logical name points to the device and directory of the root for the MDMS files.
MDMS\$SBT_ ARCHIVE	This logical name is the name of the archive used during a backup of the Oracle database.
MDMS\$SBT_ ARCHIVE_ <i>n</i>	These logical names are the names of the archives used during a backup of the Oracle database when using the duplex feature of Oracle's Recovery Manager. MDMS\$SBT_ ARCHIVE_1 is for copy 1, MDMS\$SBT_ ARCHIVE_2 is for copy 2, and so forth.
MDMS\$SBT_ CATALOG	This logical name is the name of the catalog used for the following Oracle Recovery Manager commands: • restore • validate • list backup
MDMS\$SBT_IO_BLOCK_SIZE	This logical name is the size of the block that is written on the tape volume. The default size is 65024 bytes (127 * 512 bytes). You cannot specify a value larger than 65024 bytes.
MDMS\$SBT_TRACE_LEVEL	This logical name is used by the System Backup to Tape module. It controls what is written to the Oracle trace file for SBT.
MDMS\$STARTUP_QUEUE	This logical name should point to the name of the batch queue to be used for MDMS\$STARTUP.COM.
MDMS\$SYSTEM	This logical name points to the location of MDMS utilities.
MDMS\$TCPIP_SND_PORTS	This logical name is the range of port numbers for outgoing TCP/IP connections. The defaults are ports 601 to 1023.

Logical Name	Definition and Description
MDMS\$VERSION3	This logical name is TRUE when ABS and HSM should use MDMS V3.0 or later. If ABS or HSM is not supposed to use MDMS V3.0 or later this logical will not be defined. The MDMS server should not be running if ABS and HSM is not supposed to use MDMS V3.0 or later.
MDMS\$WAIT_INITIALIZE	In order to allocate a drive when an initialize is in progress, either the process's PID should match the PID of the process doing the initialize, or 180 seconds should have passed since the start of the initialize. If the need arises, a logical interface has been provided to change the default timeout period of 180 seconds. The user needs to define a system wide logical MDMS\$WAIT_INITIALIZE to a number n, where n is the number of seconds MDMS should wait for an initialize to complete, before allowing another process to allocate the drive.

Chapter 5. ABS Logical Names

5.1. ABS Logical Names

The following logical names are entered into the system logical name table when the ABS installation procedure is complete. These names are stored in the startup file, SYS\$STARTUP:ABS\$STARTUP.COM. They are automatically entered into the system logical name table whenever the system reboots or whenever the software is invoked. The following table describes the logical names:

Logical Name ¹	Description
ABS\$CATALOG	This logical name points to the directory containing ABS catalogs.
ABS_CATALOG_CLEANUP	Defined when the catalog cleanup utility is executed.
ABS\$DATABASE	This logical name points to the Rdb/VMS database used by ABS to store policy objects. This logical is not used with V4.0, but is used during the database conversion.
ABS\$LISTINGS	This logical name points to the directory where listing files produced by ABS will reside when requested by the customer.
ABS\$LOG	This logical name points to the directory where all log files for save and restore operations are placed.
ABS\$ROOT	This logical name defines the top of the directory tree used by ABS to store its files.
ABS\$SYSTEM	This logical name points to the directory where all ABS system files and images reside.
ABS\$TEMPLATES	This logical name points to the directory where the template used to control backup agents are stored. VMS Software recommends that you do not modify these templates. The behavior of ABS in regard to its backup agents is defined by these templates.
ABS\$USSHHR	This logical points to the appropriate ABS\$USSHHR image for the current version of the operating system.

¹For portability purposes, each logical name also has a corresponding "underscore" version. For example, ABS\$CATALOG also translates as ABS_CATALOG.

5.2. DEBUG Logicals

The following logicals should only be used while debugging, as it will increase the size of the log file.

Logical Name	Description
ABS\$COORD_MAIN_TRACE	Gives information regarding the threads used by the request.
ABS\$DUMP_PIPES	If defined, gives information read from the pseudo terminal.
ABS\$DUMP_PIPES_OUTPUT	If defined, gives information written to the pseudo terminal.
ABS\$DUMP_INPUT_READY	Gives information regarding the response fed to the backup subprocess.
ABS\$DUMP_STATE_CHANGE	If defined, gives the agent states i.e. gives the state change for the subprocesses.
ABS\$MODULE_DEBUG	Gives routine names based on the path taken by the save or restore request. The level defined by the logical will determine the routines that will to be returned. It is usually defined to level 10.
ABS\$MEDIA_LOGGING	If defined, helps to log more information regarding the media_type, drive, loader, tapemarks ,and locking/unlocking of volume set.
ABS\$VOL_SET_LOGGING	If defined, gives information regarding locking/unlocking for updating volset names in storage class.
ABS\$UNLOAD_VOLUMES	If defined system wide, makes a save request to automatically unload the used volume as part of completion of the save request on that system: <pre>\$ DEFINE/SYSTEM ABS\$UNLOAD_VOLUMES TRUE</pre>

5.3. Miscellaneous Logicals

Logical Name	Description
ABS\$OPCOM_RETRIES	Allows the user to determine the number of OPCOMs to be flashed during a "no free volumes" condition. This condition is not expected during ABS restores. OPCOMs will be flashed infinitely if the logical is defined to 0. By default, OPCOMS will be flashed 10 times at an interval of 1 minute. The logical can be used only if the assist field is set in the Environment object.
ABS_SUBPROC_BASE_PRIORITY	This logical is used to change the priority of the subprocess in which the backup executes.
ABS_CATALOG_VAOE_CLEANUP	This logical must be defined system-wide if you want to execute the cleanup of the VAOE file in the Volume_Set type of catalog:

Logical Name	Description
	<pre>\$ DEFINE/SYSTEM ABS_CATALOG_VAOE_CLEANUP 1</pre> Note This user-defined logical is specific to ABS V4.4 and will be automatically removed when ABS is uninstalled. In case you want to downgrade ABS, you need to manually deassign this logical to free the space that it has occupied in the System table.
ABS_V40_LOOKUP	In ABS V4.4, the existing lookup on Volume_Set type of catalog is enhanced to use the date qualifiers effectively. This has significantly reduced the lookup time and improved the performance. However, if you still want to use the previous Lookup feature, then define the logical ABS_V40_LOOKUP, system-wide. By default, this logical is not set. <pre>\$ DEFINE/SYSTEM ABS_V40_LOOKUP 1</pre> Note This user-defined logical is specific to ABS V4.4 and will be automatically removed when ABS is uninstalled. In case you want to downgrade ABS, you need to manually deassign this logical to free the space that it has occupied in the System table.

Chapter 6. MDMS Messages

ABORT, request aborted by operator

Explanation: The request issued an OPCOM message that has been aborted by an operator. This message can also occur if no terminals are enabled for the relevant OPCOM classes on the node.

Suggested Action: Either nothing, or enable an OPCOM terminal, contact the operator, and retry.

ACCCTRLONLY, updated access control only

Explanation: You entered a **SET** command and you only had CONTROL access to the object, so only the access control information (if any) was updated.

Suggested Action: If this is what was intended, no action is needed. If you wish to update other fields in the object, you require SET access control. See your administrator.

ACCVIO, access violation

Explanation: The MDMS software caused an access violation. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis.

ALLOCDRIVE, drive !AD allocated

Explanation: The named drive was successfully allocated.

Suggested Action: None.

ALLOCDRIVEDEV, drive !AD allocated as device !AD

Explanation: The named drive was successfully allocated, and the drive may be accessed with DCL commands using the device name shown.

Suggested Action: None.

ALLOCVOLUME, volume !AD allocated

Explanation: The named volume was successfully allocated.

Suggested Action: None.

ALTSUCCESS, alternative success

Explanation: The request was successful, but extended status contains information.

Suggested Action: Examine the extended status, and retry the command as needed.

APIBUGCHECK, internal inconsistency in API

Explanation: The MDMS API (MDMS\$SHR.EXE) detected an inconsistency. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis.

APIUNEXP, unexpected error in API !AZ line !UL

Explanation: The shareable image MDMS\$SHR detected an internal inconsistency.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis.

ARCUNDEFINED, referenced archive(s) !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a archive name that does not exist. One or more of the specified archives may be undefined.

Suggested Action: Check spelling of the archive names and retry, or create the archive objects in the database.

ATTRINMAG, on-site/off-site attributes invalid for magazine-based volumes

Explanation: You attempted to specify off-site or on-site dates or locations for a volume whose placement is in a magazine. These attributes are controlled by the magazine and are not valid for individual volumes.

Suggested Action: Specify the dates and locations in the magazine object, or do not use magazines for volumes if you want the individual off-site/on-site dates to be different for each volume.

BINDVOLUME, volume !AD bound to set !AD

Explanation: The specified volume (or volume set) was successfully bound to the end of the named volume set.

Suggested Action: None.

BUGCHECK, internal inconsistency

Explanation: The server software detected an inconsistency. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis. Restart the server.

CANCELLED, request cancelled by user

Explanation: The request was cancelled by a user issuing a cancel request command.

Suggested Action: None, or retry the command.

CLEANVOL, cleaning volume loaded

Explanation: During a load of a volume, a cleaning volume was loaded.

Suggested Action: During an inventory, this message can be ignored. During a load of a requested volume, a scratch load on a drive, or an initialize command, a cleaning volume was loaded. Check the location of the cleaning volume, update the database as needed, and re-issue the command using a non-cleaning volume.

CONFLITEMS, conflicting item codes specified

Explanation: The command cannot be completed because there are conflicting item codes in the command. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis.

CREATVOLUME, volume !AD created

Explanation: The named volume was successfully created.

Suggested Action: None.

DBLOCACC, local access to database

Explanation: This node has the database files open locally.

Suggested Action: None.

DBRECERR, error !AZ !AZ record for !AZ:

Explanation: The search for a database server received an error from a remote server.

Suggested Action: Check the logfile on the remote server for more information. Check the logical name MDMS\$DATABASE_SERVERS for correct entries of database server node.

DBREMACC, access to remote database server on node !AZ

Explanation: This node has access to a remote database server.

Suggested Action: None.

DBREP, database server on node !AZ reports:

Explanation: The remote database server has reported an error condition. The next line contains additional information.

Suggested Action: Depends on the additional information.

DCLARGLSOVR, DCL extended status format, argument list overflow

Explanation: During formatting of the extended status, the number of arguments exceeded the allowable limit. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

DCLBUGCHECK, internal inconsistency in DCL

Explanation: The MDMS command line software (MDMS\$DCL.EXE) detected an inconsistency. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis.

DCSCERROR, error accessing jukebox with DCSC

Explanation: MDMS encountered an error when performing a jukebox operation. An accompanying message gives more detail.

Suggested Action: Examine the accompanying message and perform corrective actions to the hardware, the volume or the database, and optionally retry the operation.

DCSCMSG, !AZ

Explanation: This is a more detailed DCSC error message that accompanies DCSCERROR.

Suggested Action: Check the DCSC error message file.

DECNETLISEXIT, DECnet listener exited

Explanation: The DECnet listener has exited due to an internal error condition or because the user has disabled the DECNET transport for this node. The DECnet listener is the server's routine to receive requests via DECnet Phase IV and DECnet-Plus.

Suggested Action: The DECnet listener should be automatically restarted unless the DECNET transport has been disabled for this node. Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis if the transport has not been disabled by the user.

DECNETLISRUN, listening on DECnet node !AZ object !AZ

Explanation: The server has successfully started a DECnet listener. Requests can now be sent to the server via DECnet.

Suggested Action: None.

DEVNAMICM, device name item code missing

Explanation: During the allocation of a drive, the drive name was not returned by the server. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

DRIVEEXISTS, specified drive already exists

Explanation: The specified drive already exists and cannot be created.

Suggested Action: Use a set command to modify the drive, or create a new drive with a different name.

DRVACCERR, error accessing drive

Explanation: MDMS could not access the drive.

Suggested Action: Verify the VMS device name, node names, and/or group names specified in the drive record, and correct them if necessary. Verify that MDMS is running on a remote node. Check status of the drive, correct it, and retry.

DRVALRALLOC, drive is already allocated

Explanation: An attempt was made to allocate a drive that was already allocated.

Suggested Action: Wait for the drive to become deallocated, or if the drive is allocated to you, use it.

DRVEMPTY, drive is empty or volume in drive is unloaded

Explanation: The specified drive is empty, or the volume in the drive is unloaded, spun-down and inaccessible.

Suggested Action: Check status of drive, correct it, and retry.

DRVINITERR, error initializing drive on platform

Explanation: MDMS could not initialize a volume in a drive.

Suggested Action: There was a system error initializing the volume. Check the log file.

DRVINUSE, drive is currently in use

Explanation: The specified drive is already in use.

Suggested Action: Wait for the drive to free up and re-enter command, or try to use another drive.

DRVLOADED, drive is already loaded

Explanation: A drive unload appeared to succeed, but the specified volume was still detected in the drive.

Suggested Action: Check the drive and check for duplicate volume labels, or if the volume was reloaded.

DRVLOADING, drive is currently being loaded or unloaded

Explanation: The operation cannot be performed because the drive is being loaded or unloaded.

Suggested Action: Wait for the drive to become available, or use another drive. If the drive is stuck in the loading or unloading state, check for an outstanding request on the drive and cancel it. If all else fails, manually adjust the drive state.

DRVNOTALLOC, drive is not allocated

Explanation: The specified drive could not be allocated.

Suggested Action: Check again if the drive is allocated. If it is, wait until it is deallocated. Otherwise, there was some other reason the drive could not be allocated. Check the log file.

DRVNOTALLUSER, drive is not allocated to user

Explanation: You cannot perform the operation on the drive because the drive is not allocated to you.

Suggested Action: Either defer the operation or (in some cases) you may be able to perform the operation specifying a user name.

DRVNOTAVAIL, drive is not available on system

Explanation: The specified drive was found on the system, but is not available for use.

Suggested Action: Check the status of the drive and correct.

DRVNOTDEALLOC, drive was not deallocated

Explanation: MDMS could not deallocate a drive.

Suggested Action: Either the drive was not allocated or there was a system error deallocating the drive. Check the log file.

DRVNOTFOUND, drive not found on system

Explanation: The specified drive cannot be found on the system.

Suggested Action: Check that the OpenVMS device name, node names and/or group names are correct for the drive. Verify MDMS is running on a remote node. Re-enter command when corrected.

DRVNOTSPEC, drive not specified or allocated to volume

Explanation: When loading a volume a drive was not specified, and no drive has been allocated to the volume.

Suggested Action: Retry the operation and specify a drive name.

DRVREMOTE, drive is remote

Explanation: The specified drive is remote on a node where it is defined to be local.

Suggested Action: Check that the OpenVMS device name, node names, and/or group names are correct for the drive. Verify MDMS is running on a remote node. Re-enter command when corrected.

DRVSINUSE, all drives are currently in use

Explanation: All of the drives matching the selection criteria are currently in use.

Suggested Action: Wait for a drive to free up and re-enter command.

DRVUNDEFINED, referenced drive !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a drive name that does not exist.

Suggested Action: Check spelling of the drive name and retry, or create the drive object in the database.

ENVUNDEFINED, referenced environments !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a environment name that does not exist. One or more of the specified environments may be undefined.

Suggested Action: Check spelling of the environment names and retry, or create the environment objects in the database.

ERROR, error

Explanation: A general internal MDMS error occurred.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

EXECOMFAIL, execute command failed, see log file for more explanation

Explanation: While trying to execute a command during scheduled activities, a system service called failed.

Suggested Action: Check the log file for the failure code from the system server call.

EXIT, MDMS server exiting with fatal error, restarting

Explanation: The MDMS server has encountered a fatal error and is exiting. The server will be restarted.

Suggested Action: Report the incident to VMS Software.

EXSCHED, internal schedules are inoperable; external scheduler in use

Explanation: You have created or modified an MDMS schedule object. This is allowed, but since the domain scheduler type is set up to an external scheduler product, this schedule object will never be executed.

Suggested Action: If you are not planning to change the scheduler type to INTERNAL or EXTERNAL, you should modify the associated save or restore request to use a standard frequency or an explicit frequency.

EXTRAVOL, extra volume(s) processed

Explanation: One or more volumes unknown to MDMS have been processed by this command.

Suggested Action: See next message lines for more details. Use MDMS or jukebox utility programs (MRU or CARTRIDGE) to correct the problem.

FAILALLOCDRV, failed to allocate drive

Explanation: Failed to allocate drive.

Suggested Action: The previous message is the error that caused the failure.

FAILCONSVR, failed connection to server

Explanation: The connection to an MDMS server either failed or could not be established. See additional message lines and/or check the server's logfile.

Suggested Action: Depends on additional information.

FAILCONSVRD, failed connection to server via DECnet

Explanation: The DECnet connection to an MDMS server either failed or could not be established. See additional message lines and/or check the server's logfile.

Suggested Action: Depends on additional information.

FAILCONSVRT, failed connection to server via TCP/IP

Explanation: The TCP/IP connection to an MDMS server either failed or could not be established. See additional message lines and/or check the server's logfile.

Suggested Action: Depends on additional information.

FAILCREATE, failed to create !AZ

Explanation: The reported file or object could not be created. The next line contains additional information.

Suggested Action: Depends on the additional information.

FAILDEALLOCDRV, failed to deallocate drive

Explanation: Failed to deallocate drive.

Suggested Action: The previous message is the error that caused the failure.

FAILDELETE, failed to delete !AZ

Explanation: The reported file or object could not be deleted. The next line contains additional information.

Suggested Action: Depends on the additional information.

FAILEDMNTVOL, failed to mount volume

Explanation: MDMS was unable to mount the volume.

Suggested Action: The error above this contains the error that cause the volume not to be mounted.

FAILICRES, failed item code restrictions

Explanation: The command cannot be completed because there are conflicting item codes in the command. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

FAILINIEXTSTAT, failed to initialize extended status buffer

Explanation: The API could not initialize the extended status buffer. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

FAILLOOKUP, failed to lookup !AZ

Explanation: The reported file or object could not be looked up. The next line contains additional information.

Suggested Action: Depends on the additional information.

FAILURE, fatal error

Explanation: The MDMS server encountered a fatal error during the processing of a request.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

FILOPNERR, file !AZ could not be opened

Explanation: An MDMS database file could not be opened.

Suggested Action: Check the server's logfile for more information.

FIRSTVOLUME, specified volume is first in set

Explanation: The specified volume is the first volume in a volume set.

Suggested Action: You cannot deallocate or unbind the first volume in a volume set. However, you can unbind the second volume and then deallocate the first, or unbind and deallocate the entire volume set.

FUNCFALLED, function !AZ failed with

Explanation: An internal call to a system function has failed. The following lines identify the function called and the failure status.

Suggested Action: Depends on information following this message.

GRPUNDEFINED, referenced group(s) !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a group name that does not exist. One or more of the specified groups may be undefined.

Suggested Action: Check spelling of the group names and retry, or create the group objects in the database.

ILLEGALOP, illegal move operation

Explanation: You attempted to move a volume within a DCSC jukebox, and this is not supported.

Suggested Action: None.

INCOMFREQ, incompatible frequency for !AZ !AZ

Explanation: After changing the domain scheduler type, MDMS has determined that this save or restore request has a frequency that is incompatible with the new scheduler type. The frequencies that are not valid for the given scheduler types are:

- INTERNAL and EXTERNAL – Explicit
- DECSCHEDULER and SCHEDULER – Custom

Suggested Action: Modify the frequency to a valid one for this scheduler type.

INCOMPATMED, volume's media type incompatible with the drive

Explanation: The media type for the volume is incompatible with the media types for the drive on a load operation.

Suggested Action: Verify that the volume can be physically loaded and used in the specified drive. If not, select another drive. If so, then add the volume's media type to the drive or otherwise align the media types of the volume and the drive.

INCOMPATOPT, incompatible options specified

Explanation: You entered a command with incompatible options.

Suggested Action: Examine the command documentation and re-enter with allowed combinations of options.

INCOMPATTR, attributes incompatible with archive type

Explanation: You attempted to create or set an attribute that is incompatible with the archive type. The following attributes are incompatible for archive types:

- DISK: CONSOLIDATION, DRIVES, MEDIA_TYPE, POOL, VOLUME_SETS
- TAPE: DESTINATION

Suggested Action: Do not specify these attributes if they are incompatible with the archive type.

INCOMPATVOL, volume is incompatible with volumes in set

Explanation: You cannot bind the volume to the volume set because some of the volume's attributes are incompatible with the volumes in the volume set.

Suggested Action: Check that the new volume's media type, on-site location, and off-site location are compatible with those in the volume set. Adjust attributes and retry, or use another volume with compatible attributes.

INSCMDPRIV, insufficient privilege to execute request

Explanation: You do not have sufficient privileges to enter the request.

Suggested Action: Contact your system administrator and request additional privileges, or give yourself privileges and retry.

INSOPTPRIV, insufficient privilege for request option

Explanation: You do not have sufficient privileges to enter a privileged option of this request.

Suggested Action: Contact your system administrator and request additional privileges, or give yourself privileges and retry. Alternatively, retry without using the privileged option.

INSSHOWPRIV, some volumes not shown due to insufficient privilege

Explanation: Not all volumes were shown because of restricted privilege.

Suggested Action: None, if you just want to see volumes you own. You need MDMS_SHOW_ALL privilege to see all volumes.

INSSVRPRV, insufficient server privileges

Explanation: The MDMS server is running with insufficient privileges to perform system functions.

Suggested Action: Refer to the *VSI Archive Backup System for OpenVMS Installation Guide* [<https://docs.vmssoftware.com/vsi-abs-installation-guide/>] to determine the required privileges. Contact your system administrator to add these privileges in the MDMS\$SERVER account.

INTBUFOVR, internal buffer overflow

Explanation: The MDMS software detected an internal buffer overflow. This an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis. Restart the server.

INTINVMSG, internal invalid message

Explanation: An invalid message was received by a server. This could be due to a network problem or, a remote non-MDMS process sending messages in error or, an internal error.

Suggested Action: If the problem persists and no non-MDMS process can be identified then provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

INTSCHEDULE, cannot modify or delete internal schedule

Explanation: You attempted to modify or delete a schedule object that was internally generated for a save or restore request. This is not allowed.

Suggested Action: Modify or delete the associated save or restore request instead, and the schedule will be updated accordingly.

INVABSTIME, invalid absolute time

Explanation: The item list contained an invalid absolute date and time. Time cannot be earlier than 1-Jan-1970 00:00:00 and cannot be greater than 7-Feb-2106 06:28:15.

Suggested Action: Check that the time is between these two times.

INVALIDRANGE, invalid volume ID or invalid range specified

Explanation: The specified volume ID, volume range, slot range, or space range is invalid.

Suggested Action: A volume ID may contain up to 6 characters. A volume range may contain up to 1000 volume IDs where the first 3 characters must be alphabetic and the last 3 may be alphanumeric. Only the numeric portions may vary in the range. Examples are ABC000-ABC999 or ABCD01-ABCD99. A slot range can contain up to 1000 slots and must be numeric. Also, all slots in the range must be less than the slot count for the jukebox or magazine.

For example, 0-255 for a slot count of 256. A space range can contain up to 1000 spaces where the first and last spaces must have the same number of characters. Spaces must be within the range defined for the location. Examples are 000-999 or Space A1-Space C9.

INVCONSOLVAL, invalid value for consolidation savesets or volumes

Explanation: You specified an invalid value for consolidation savesets or volumes.

Suggested Action: Use a value in the range 0 to maximum integer.

INVDBSVRLIS, invalid database server search list

Explanation: The logical name MDMS\$DATABASE_SERVERS contains invalid network node names or is not defined.

Suggested Action: Correct the node names in the logical name MDMS\$DATABASE_SERVERS in file MDMS\$SYSTARTUP.COM. Redefine the logical name in the current system. Then, start the server.

INVDELSTATE, object is in invalid state for delete

Explanation: The specified object cannot be deleted because its state indicates it is being used.

Suggested Action: Defer deletion until the object is no longer being used, or otherwise change its state and retry.

INVELTATIME, invalid delta time

Explanation: The item list contained an invalid delta time.

Suggested Action: Check that the item list has a correct delta time.

INVDFULLNAM, invalid DECnet fullname

Explanation: A node full name for a DECnet-Plus node specification has an invalid syntax.

Suggested Action: Correct the node name and retry.

INVDRVCOUNT, invalid value for drive count, use 1-32

Explanation: You specified an invalid value for drive count.

Suggested Action: Use a value in the range 1-32.

INTEXTSTS, invalid extended status item desc/buffer

Explanation: The error cannot be reported in the extended status item descriptor. This error can be caused by one of the following:

- Not being able to read any one of the item descriptors in the item list
- Not being able to write to the buffer in the extended status item descriptor
- Not being able to write to the return length in the extended status item descriptor
- Not being able to initialize the extended status buffer

Suggested Action: Check for any of the errors stated above in your program and fix the error.

INVFREQUENCY, invalid frequency for domain scheduler type

Explanation: You specified an invalid save or restore frequency the scheduler type specified in the domain.

Invalid combinations include: CUSTOM, with NONE, DECSCHEDULER, SCHEDULER or LOCAL EXPLICIT, with NONE, INTERNAL, EXTERNAL, or SINGLE.

Suggested Action: Specify a valid frequency for the scheduler type specified in the domain.

INVINITOPT, invalid initialize options specified

Explanation: You attempted initialize volumes in a jukebox by specifying a slot range and the jukebox is not a vision-equipped, MRD-controlled jukebox.

Suggested Action: Specify a volume range instead of a slot range to initialize volumes in a DCSC jukebox or an MRD jukebox without a vision system.

INVITCODE, invalid item code for this function

Explanation: The item list had an invalid item code. The problem could be one of the following:

- Item codes do not meet the restrictions for that function.
- An item code cannot be used in this function.

Suggested Action: Refer to the API specification to find out which item codes are restricted for each function and which item codes are allowed for each function.

INVITDESC, invalid item descriptor, index !@UL

Explanation: The item descriptor is in error. The previous message gives the error. Included is the index of the item descriptor in the item list.

Suggested Action: Refer to the index number and the previous message to indicate the error and which item descriptor is in error.

INVITLLENGTH, invalid item list buffer length

Explanation: The item list buffer length is zero. The item list buffer length cannot be zero for any item code.

Suggested Action: Refer to the API specification to find an item code that would be used in place of an item code that has a zero buffer length.

INVMAXSAVES, invalid value for maximum saves, use 1-36

Explanation: You specified an invalid value for maximum saves.

Suggested Action: Use a value in the range 1-36.

INVMEDIATYPE, media type is invalid or not supported by volume

Explanation: The specified volume supports multiple media types where a single media type is required, or the volume does not support the specified media type.

Suggested Action: Re-enter the command specifying a single media type that is already supported by the volume.

INVMSG, invalid message via !AZ

Explanation: An invalid message was received MDMS software. This could be due to a network problem or, a non-MDMS process sending messages in error or, an internal error.

Suggested Action: If the problem persists and no non-MDMS process can be identified then provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

INVNODNAM, invalid node name specification

Explanation: A node name for a DECnet Phase IV node specification has an invalid syntax.

Suggested Action: Correct the node name and retry.

INVPORTS, invalid port number specification

Explanation: The MDMS server did not start up because the logical name MDMS\$TCPIP_SND_PORTS in file MDMS\$SYSTARTUP.COM specifies an illegal port number range. A legal port number range is of the form "*low-port-number-high-port-number*".

Suggested Action: Correct the port number range for the logical name MDMS\$TCPIP_SND_PORTS in file MDMS\$SYSTARTUP.COM. Then, start the server.

INVPOSITION, invalid jukebox position

Explanation: The position specified is invalid.

Suggested Action: Position is only valid for jukeboxes with a topology defined. Check that the position is within the topology ranges, correct, and retry. For example, **/POSITION= (1, 2, 1)**.

INVRETDAYS, invalid retention days specified

Explanation: You entered an invalid value for the retention days. Valid values are 0 to 9999 days. If you wish for no expiration of volumes, specify **/NOEXPIRATION_DATE**.

Suggested Action: Enter a value between 0 and 9999.

INVRETRY, invalid value for retry count or interval

Explanation: You specified an invalid value for either or both the retry count or interval. In addition, it is invalid to specify an interval with a retry limit of zero or nolimit.

Suggested Action: Use values within the following ranges:

- **RETRY_LIMIT:** 0–10000 or NOLIMIT
- **INTERVAL:** 00:01:00–01:00:00 (1–60 mins)

INVRETRYINTERVAL, invalid value for retry interval

Explanation: You specified an invalid value for retry interval. In addition, it is invalid to specify an interval with a retry limit of zero.

Suggested Action: Use a value within the following range only if retry limit is non-zero: 00:01:00–01:00:00 (1–60 minutes).

INVRETRYLIMIT, invalid value for retry limit

Explanation: You specified an invalid value for retry limit.

Suggested Action: Use a value in the range 0 to maximum integer or use **/NORETRY_LIMIT**.

INVSCHEDENUM, invalid scheduling translation defined

Explanation: An invalid parameter translation was entered for a scheduling option.

Suggested Action: Report the incident to VMS Software.

INVSCHEDOPT, invalid schedule options entered

Explanation: You entered invalid schedule date/time options for a schedule object. The following values are allowed:

- DATES: List of values or ranges, values 1–31
- DAYS: List of values or ranges, values MON–SUN
- MONTHS: List of values or ranges, values JAN–DEC
- TIMES: List of values, 00:00–23:59
- INCLUDE: List of dates 01-Jan-yyyy–31-Dec-yyyy
- EXCLUDE: List of dates 01-Jan-yyyy–31-Dec-yyyy

The yyyy for INCLUDE and EXCLUDE must be between the current year and up to 9 years into the future (e.g. 2001-2010). If omitted, the current year is used.

Suggested Action: Re-enter the command with valid values.

INVSCHEDPARAM, invalid scheduling parameter defined

Explanation: An invalid parameter was entered for a scheduling option.

Suggested Action: Report the incident to VMS Software.

INVSELECT, invalid selection criteria

Explanation: The selection criteria specified on an allocate command are invalid.

Suggested Action: Check the command with the documentation and re-enter with a valid combination of selection criteria.

INVSLOT, invalid slot or slot range specified

Explanation: The slot or slot range specified when moving volumes into a magazine or jukebox was invalid, or the specified slots were already occupied.

Suggested Action: Specify valid empty slots and re-enter.

INVSLOT RANGE, invalid slot range

Explanation: The slot range was invalid. It must be of the form: 1–100 1,100–200,300–400 The only characters allowed are comma, dash, and numbers (0-9).

Suggested Action: Check that you are using the correct form.

INVSPACE, invalid space or space range specified

Explanation: The space or space range specified when moving volumes into a location was invalid.

Suggested Action: Specify valid spaces already defined for the location, or specify a space range for the location

INVSRCDEST, invalid source or destination for move

Explanation: Either the source or destination of a move operation was invalid (does not exist).

Suggested Action: If the destination is invalid, enter a correct destination and retry. If a source is invalid, either create the source or correct the current placement of the affected volumes or magazines.

INVSTATE, volume !AZ is in an invalid state for initialization

Explanation: The volume loaded in the drive for initialization was either allocated or in the transition state and cannot be initialized.

Suggested Action: Either the wrong volume was loaded, or the requested volume was in an invalid state. If the wrong volume was loaded, perform an inventory on the jukebox and retry. If the volume is allocated or in transition, you should not try to initialize the volume.

INVTFULLNAM, invalid TCP/IP fullname

Explanation: A node full name for a TCP/IP node specification has an invalid syntax.

Suggested Action: Correct the node name and retry.

INVTPOLOGY, invalid jukebox topology

Explanation: The specified topology for a jukebox is invalid.

Suggested Action: Check topology definition; the towers must be sequentially increasing from 0; there must be a face, level, and slot definition for each tower. For example:

```
/TOPOLOGY=(TOWER=(0,1,2), FACES=(8,8,8), -  
LEVELS=(2,3,2), SLOTS=(13,13,13))
```

INVVOLPLACE, invalid volume placement for operation

Explanation: The volume has an invalid placement for a load operation.

Suggested Action: Re-enter the command and use the move option.

INVOLSTATE, volume in invalid state for operation

Explanation: The operation cannot be performed on the volume because of the volume state does not allow it.

Suggested Action: Defer the operation until the volume changes state. If the volume is stuck in a transient state (e.g. moving), check for an outstanding request and cancel it. If all else fails, manually change the state.

JUKEBOXEXISTS, specified jukebox already exists

Explanation: The specified jukebox already exists and cannot be created.

Suggested Action: Use a set command to modify the jukebox, or create a new jukebox with a different name.

JUKENOTINIT, jukebox could not be initialized

Explanation: An operation on a jukebox failed because the jukebox could not be initialized.

Suggested Action: Check the control, robot name, node name, and group name of the jukebox, and correct as needed. Check access path to jukebox (HSJ etc), correct as needed. Verify MDMS is running on a remote node. Then retry operation.

JUKETIMEOUT, timeout waiting for jukebox to become available

Explanation: MDMS timed out waiting for a jukebox to become available. The timeout value is 10 minutes.

Suggested Action: If the jukebox is in heavy use, try again later. Otherwise, check requests for a hung request and cancel it. If all else fails, set the jukebox state to available.

JUKEUNAVAIL, jukebox is currently unavailable

Explanation: The jukebox is disabled.

Suggested Action: Re-enable the jukebox.

JUKUNDEFINED, referenced jukebox !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a jukebox name that does not exist.

Suggested Action: Check spelling of the jukebox name and retry, or create the jukebox object in the database.

LOCATIONEXISTS, specified location already exists

Explanation: The specified location already exists and cannot be created.

Suggested Action: Use a set command to modify the location, or create a new location with a different name.

LOCUNDEFINED, referenced location !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a location name that does not exist.

Suggested Action: Check spelling of the location name and retry, or create the location object in the database.

LOGRESET, Log file !AZ by !AZ on node !AZ

Explanation: The server logfile has been closed and a new version has been created by a user.

Suggested Action: None.

MAGAZINEEXISTS, specified magazine already exists

Explanation: The specified magazine already exists and cannot be created.

Suggested Action: Use a set command to modify the magazine, or create a new magazine with a different name.

MAGUNDEFINED, referenced magazine !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a magazine name that does not exist.

Suggested Action: Check spelling of the magazine name and retry, or create the magazine object in the database.

MBLISEXIT, mailbox listener exited

Explanation: The mailbox listener has exited due to an internal error condition. The mailbox listener is the server's routine to receive local user requests through mailbox MDMS\$MAILBOX.

Suggested Action: The mailbox listener should be automatically restarted. Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

MBLISRUN, listening on mailbox !AZ logical !AZ

Explanation: The server has successfully started the mailbox listener. MDMS commands can now be entered on this node.

Suggested Action: None.

MEDIATYPEEXISTS, specified media type already exists

Explanation: The specified media type already exists and cannot be created.

Suggested Action: Use a set command to modify the media type, or create a new media type with a different name.

MEDUNDEFINED, referenced media type(s) !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a media type that does not exist. One or more of the specified media types may be undefined.

Suggested Action: Check spelling of the media types and retry, or create the media type objects in the database.

MOVEINCOMPL, move is incomplete

Explanation: When moving volumes into and out of a jukebox, some of the volumes were not moved.

Suggested Action: Check that there are enough empty slots in the jukebox when moving in and retry. On a move out, examine the cause of the failure and retry.

MRDERROR, error accessing jukebox with MRD

Explanation: MDMS encountered an error when performing a jukebox operation. An accompanying message gives more detail.

Suggested Action: Examine the accompanying message and perform corrective actions to the hardware, the volume or the database, and optionally retry the operation.

MRDMSG, !AZ

Explanation: This is a more detailed MRD error message which accompanies MRDERROR.

Suggested Action: Check the MRU error message file.

NOACCESS, no user access to object for operation

Explanation: You attempted to perform an operation on an object for which you have no access.

Suggested Action: You need an authorized user to add you to the access control list, otherwise you cannot perform the requested operation.

NOBINDSELF, volume is already in volume set

Explanation: You cannot bind this volume into this volume set because it already a member of the volume set.

Suggested Action: Use another volume.

NOCHANGES, no attributes were changed in the database

Explanation: Your set command did not change any attributes in the database because the attributes you entered were already set to those values.

Suggested Action: Double-check your command, and re-enter if necessary. Otherwise the database is already set to what you entered.

NOCHANGESOBJ, no attributes were changed for !AZ !AZ

Explanation: Your set command did not change any attributes in the database because the attributes you entered were already set to those values. The message indicates which object was not changed.

Suggested Action: Double-check your command, and re-enter if necessary. Otherwise the database is already set to what you entered.

NOCHECK, drive not accessible, check not performed

Explanation: The specified drive could not be physically accessed and the label check was not performed. The displayed attributes are taken from the database.

Suggested Action: Verify the VMS device name, node name or group name in the drive object. Check availability on system. Verify MDMS is running on a remote node. Determine the reason the drive was not accessible, fix it and retry.

NODBACC, no access to database server

Explanation: This server has no access to a database server.

Suggested Action: Verify the setting of logical name MDMS\$DATABASE_SERVERS. Check each node listed for connectivity and database access status using the following command:

```
$ MDMS SHOW SERVER/NODES=node-name
```

Check the servers logfiles for more information.

NODCSC, DCSC not running

Explanation: DCSC has not been started.

Suggested Action: Execute command procedure SYS\$STARTUP:DCSC\$STARTUP.COM and retry command.

NODEDISABLED, node disabled

Explanation: The server failed to start up because it is disabled in the database.

Suggested Action: If necessary, correct the setting and start the server again.

NODEEXISTS, specified node already exists

Explanation: The specified node already exists and cannot be created.

Suggested Action: Use a **SET** command to modify the node, or create a new node with a different name.

NODENOPRIV, node is not privileged to access database server

Explanation: A remote server access failed because the user making the DECnet connection is not MDMS\$SERVER or the remote port number is not less than 1024.

Suggested Action: Verify with DCL command **SHOW PROCESS** that the remote MDMS server is running under a username of MDMS\$SERVER. Alternatively, verify that logical name MDMS\$TCPIP_SND_PORTS on the remote server node specifies a port number range between 0-1023.

NODENOTENA, node not in database or not fully enabled

Explanation: The server was not allowed to start up because there is no such node object in the database or its node object in the database does not specify all network full names correctly.

Suggested Action: For a node running DECnet Phase IV, the node name has to match logical name SYS\$NODE on that node.

For a node running DECnet-Plus, the node's DECNET_PLUS_FULLNAME has to match the logical name SYS\$NODE_FULLNAME on that node.

For a node running TCP/IP, the node's TCPIP_FULLNAME has to match the full name combined from logical names *INET_HOST and *INET_DOMAIN.

NODENOTINDB, no node object with !AZ name !AZ in database

Explanation: The current server could not find a node object in the database with a matching DECnet Phase IV or DECnet-Plus or TCP/IP node full name.

Suggested Action: Use the following command to see the exact naming of the server's network names:

```
$ MDMS SHOW SERVER/NODES=node-name
```

Correct the entry in the database and restart the server.

NODRIVES, no drives match selection criteria

Explanation: When allocating a drive, none of the drives match the specified selection criteria.

Suggested Action: Check spelling and re-enter command with valid selection criteria.

NODRVACC, access to drive disallowed

Explanation: You attempted to allocate, load or unload a drive from a node that is not allowed to access it.

Suggested Action: The access field in the drive object allows local, remote, or all access, and your attempted access did not conform to the attribute. Use another drive.

NODRVSAVAIL, no drives are currently available

Explanation: All of the drives matching the selection criteria are currently in use or otherwise unavailable.

Suggested Action: Check to see if any of the drives are disabled or inaccessible. Re-enter command when corrected.

NODRVSGRP, no drives in the specified group were found

Explanation: When allocating a drive, no drives on nodes in the specified group were found.

Suggested Action: Check group name and retry command.

NODRVSJUKE, no drives in the specified jukebox were found

Explanation: When allocating a drive, no drives in the specified jukebox were found.

Suggested Action: Check jukebox name and retry command.

NODRVSLOC, no drives in the specified location were found

Explanation: When allocating a drives, no drives in the specified location were found.

Suggested Action: Check location name and retry command.

NODRVSMED, no drives with the specified media type were found

Explanation: When allocating a drive, no drives with the specified media type were found.

Suggested Action: Check media type and retry command, or specify the media type for more drives.

NODRVSNOD, no drives on the specified node were found

Explanation: When allocating a drive, no drives on the specified node were found.

Suggested Action: Check the node name and retry command.

NODRVSVOL, no drives that can support the specified volume were found

Explanation: When allocating a drive, no drives that could support the specified volume were found.

Suggested Action: Check the volume ID and retry command, or check and adjust volume attributes to match a valid drive.

NODUNDEFINED, referenced node(s) !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a node name that does not exist. One or more of the specified nodes may be undefined.

Suggested Action: Check spelling of the node names and retry, or create the node objects in the database.

NOFIELDS, no fields specified for report

Explanation: An **MDMS REPORT VOLUME** command was entered with no fields to select or display.

Suggested Action: Enter at least one field for the report.

NOINCLUDE, selection attributes not set with no include data

Explanation: You specified one or more of the following attributes that are not valid unless an include specification is present: **DATA_TYPE**, **INCREMENTAL**, **NODES**, or **GROUPS**.

The save or restore object was updated, but selection attributes were not set.

Suggested Action: These attributes are applicable only when an **INCLUDE** statement is present. Re-enter the command with an **/INCLUDE** qualifier.

NOINCLUDES, no include specification for selection

Explanation: A save or restore object had some selection attributes specified, but no include file specification.

The following attributes require an include specification: **DATA_TYPE**, **INCREMENTAL**, **NODES**, or **GROUPS**.

Suggested Action: Re-enter the command with an include specification.

NOINTSCHED, internal scheduling not enabled

Explanation: You attempted to create a schedule object but the domain's scheduler option is set to an external scheduler. The MDMS schedule object is valid only with scheduler options **INTERNAL**, **EXTERNAL**, and **SINGLE_SCHEDULER**.

Suggested Action: Schedule your request using the specified external scheduler product and interface.

NOJUKEACC, access to jukebox disallowed

Explanation: You attempted to use a jukebox from a node that is not allowed to access it.

Suggested Action: The access field in the jukebox object allows local, remote, or all access, and your attempted access did not conform to the attribute. Use another jukebox.

NOJUKESPEC, jukebox required on vision option

Explanation: The jukebox option is missing on a create volume request with the vision option.

Suggested Action: Re-enter the request and specify a jukebox name and slot range.

NOLICENSE, your current license does not support this operation

Explanation: The requested operation is not licensed.

Suggested Action: Use an alternative mechanism to perform the operation. If this is not possible, you cannot perform the operation with your current license. You may purchase an upgrade ABS license to enable full ABS functionality. Contact VMS Software for details.

NOMAGAZINES, no magazines match selection criteria

Explanation: On a move magazine request using the schedule option, no magazines were scheduled to be moved.

Suggested Action: None.

NOMAGSMOVED, no magazines were moved

Explanation: No magazines were moved for a move magazine operation. An accompanying message gives a reason.

Suggested Action: Check the accompanying message, correct, and retry.

NOMEDIATYPE, no media type specified when required

Explanation: An allocation for a volume based on node, group, or location also requires the media type to be specified.

Suggested Action: Re-enter the command with a media type specification.

NOMEMORY, not enough memory

Explanation: The MDMS server failed to allocate enough virtual memory for an operation. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis. Restart the server.

NOOBJECTS, no such objects currently exist

Explanation: On a show command, there are no such objects currently defined.

Suggested Action: None.

NOPARAM, required parameter missing

Explanation: A required input parameter to a request or an API function was missing.

Suggested Action: Re-enter the command with the missing parameter, or refer to the API specification for required parameters for each function.

NOPOOLSPEC, no free volumes with no pool or your default pool were found

Explanation: When allocating a volume, no free volumes that do not have a pool defined or that are in your default pool were found.

Suggested Action: Add a pool specification to the command, or define more free volumes with no pool or your default pool.

NORANGESUPP, slot or space ranges not supported with volset option

Explanation: On a set volume, you entered the volset option and specified either a slot range or space range.

Suggested Action: If you want to assign slots or spaces to volumes directly, do not use the volset option.

NORECVPORTS, no available receive port numbers for incoming connections

Explanation: The MDMS could not start the TCP/IP listener because none of the receive ports specified with this node's TCPIP_FULLNAME are currently available.

Suggested Action: Use a suitable network utility to find a free range of TCP/IP ports which can be used by the MDMS server. Use the **MDMS SET NODE** command to specify the new range with the **/TCP_IP_FULLNAME** then restart the server.

NOREMCONNECT, unable to connect to remote node

Explanation: The server could not establish a connection to a remote node. See the server's logfile for more information.

Suggested Action: Depends on information in the logfile.

NOREQUESTS, no such requests currently exist

Explanation: No requests exist on the system.

Suggested Action: None.

NORESEFN, not enough event flags

Explanation: The server ran out of event flags. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis. Restart the server.

NORIGHTS, no rights are shown

Explanation: When showing a domain, the rights are not shown because you don't have privilege to see the rights.

Suggested Action: Nothing. To see rights you need MDMS_SHOW_RIGHTS.

NOSCHEDULE, schedule object invalid for scheduler type or frequency

Explanation: You specified a schedule object for a non-custom frequency or for an external scheduler option. A schedule object can only be specified for frequency CUSTOM with domain scheduler type of INTERNAL, EXTERNAL, or SINGLE.

Suggested Action: Do not specify a schedule name.

NOSCRATCH, scratch loads not supported for jukebox drives

Explanation: You attempted a load drive command for a jukebox drive.

Suggested Action: Scratch loads are not supported for jukebox drives. You must use the load volume command to load volumes in jukebox drives.

NOSENDPORTS, no available send port numbers for outgoing connection

Explanation: The server could not make an outgoing TCP/IP connection because none of the send ports specified for the range in logical name MDMS\$TCPIP_SND_PORTS are currently available.

Suggested Action: Use a suitable network utility to find a free range of TCP/IP ports which can be used by the MDMS server. Change the logical name MDMS\$TCPIP_SND_PORTS in file MDMS\$SYSTARTUP.COM. Then, restart the server.

NOSLOT, not enough slots defined for operation

Explanation: The command cannot be completed because there are not enough slots specified in the command or because there are not enough empty slots in the jukebox.

Suggested Action: If the jukebox is full, move some other volumes out of the jukebox and retry. If there are not enough slots specified in the command, re-enter with a larger slot range.

NOSTATUS, no status defined

Explanation: An uninitialized status has been reported. This an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis.

NOSUCHDEST, specified destination does not exist

Explanation: In a move command, the specified destination does not exist.

Suggested Action: Check spelling or create the destination as needed.

NOSUCHDRIVE, specified drive does not exist

Explanation: The specified drive does not exist.

Suggested Action: Check spelling or create drive as needed.

NOSUCHGROUP, specified group does not exist

Explanation: The specified group does not exist.

Suggested Action: Check spelling or create group as needed.

NOSUCHINHERIT, specified inherited object does not exist

Explanation: On a create of an object, the object specified for inherit does not exist.

Suggested Action: Check spelling or create the inherited object as needed.

NOSUCHJUKEBOX, specified jukebox does not exist

Explanation: The specified jukebox does not exist.

Suggested Action: Check spelling or create jukebox as needed.

NOSUCHLOCATION, specified location does not exist

Explanation: The specified location does not exist.

Suggested Action: Check spelling or create location as needed.

NOSUCHMAGAZINE, specified magazine does not exist

Explanation: The specified magazine does not exist.

Suggested Action: Check spelling or create magazine as needed.

NOSUCHMEDIATYPE, specified media type does not exist

Explanation: The specified media type does not exist.

Suggested Action: Check spelling or create media type as needed.

NOSUCHNODE, specified node does not exist

Explanation: The specified node does not exist.

Suggested Action: Check spelling or create node as needed.

NOSUCHOBJECT, specified object does not exist

Explanation: The specified object does not exist.

Suggested Action: Check spelling or create the object as needed.

NOSUCHPOOL, specified pool does not exist

Explanation: The specified pool does not exist.

Suggested Action: Check spelling or create pool as needed.

NOSUCHREQUESTID, specified request does not exist

Explanation: The specified request does not exist on the system.

Suggested Action: Check the request id again, and re-enter if incorrect.

NOSUCHUSER, no such user on system

Explanation: The username specified in the command does not exist.

Suggested Action: Check spelling of the username and re-enter.

NOSUCHVOLUME, specified volume(s) do not exist

Explanation: The specified volume or volumes do not exist.

Suggested Action: Check spelling or create volumes as needed.

NOSVRACCOUNT, username !AZ does not exist

Explanation: The server cannot start up because the username MDMS\$SERVER is not defined in file SYSUAF.DAT.

Suggested Action: Enter the username of MDMS\$SERVER (see the [VSI Archive Backup System for OpenVMS Installation Guide](https://docs.vmssoftware.com/vsi-abs-installation-guide/) [https://docs.vmssoftware.com/vsi-abs-installation-guide/] for account details) and then start the server.

NOSVRMB, no server mailbox or server not running

Explanation: The MDMS server is not running on this node or the server is not servicing the mailbox via logical name MDMS\$MAILBOX.

Suggested Action: Use the MDMS\$STARTUP procedure with parameter RESTART to restart the server. If the problem persists, check the server's logfile and file SYSS\$MANAGER:MDMS\$SERVER.LOG for more information.

NOSYMBOLS, symbols not supported for multiple volumes

Explanation: An **MDMS SHOW VOLUME/SYMBOLS** command was entered for multiple volumes. The **/SYMBOLS** qualifier is only supported for a single volume.

Suggested Action: Re-enter the command with a single volume ID, or do not use the **/SYMBOLS** qualifier.

NOTALLOCUSER, volume is not allocated to user

Explanation: You cannot perform the operation on the volume because the volume is not allocated to you.

Suggested Action: Either use another volume, or (in some cases) you may be able to perform the operation specifying a user name.

NOTSCHEDULED, specified save or restore is not scheduled for execution

Explanation: The save or restore request did not contain enough information to schedule the request for execution. The request requires the definition of an archive, an environment and a start time.

Suggested Action: If you wish this request to be scheduled, enter an **MDMS SET SAVE** or **MDMS SET RESTORE** command and enter the required information.

NOUNALLOCDRV, no unallocated drives found for operation

Explanation: On an initialize volume request, MDMS could not locate an unallocated drive for the operation.

Suggested Action: If you had allocated a drive for the operation, deallocate it, and retry. If all drives are currently in use, retry the operation later.

NOVOLSJUKE, no free volumes in the specified jukebox were found

Explanation: When allocating a volume, no free volumes in the specified jukebox were found.

Suggested Action: Check jukebox name and retry command, or move some free volumes into the jukebox.

NOVOLSLOC, no free volumes in the specified location were found

Explanation: When allocating a volume, no free volumes in the specified location were found.

Suggested Action: Check location name and retry command, or move some free volumes into the location.

NOVOLSMED, no free volumes with the specified media type were found

Explanation: When allocating a volume, no free volumes with the specified media type were found.

Suggested Action: Check media type and retry command, or specify the media type for more free volumes.

NOVOLSMOVED, no volumes were moved

Explanation: No volumes were moved for a move volume operation. An accompanying message gives a reason.

Suggested Action: Check the accompanying message, correct, and retry.

NOVOLSPPOOL, no free volumes in the specified pool were found

Explanation: When allocating a volume, no free volumes in the specified pool were found.

Suggested Action: Check pool name and retry command, or specify the pool for more free volumes (add them to the pool).

NOVOLSPROC, no volumes were processed

Explanation: In a create, set or delete volume command, no volumes were processed.

Suggested Action: Check the volume identifiers and re-enter command.

NOVOLSVOL, no free volumes matching the specified volume were found

Explanation: When allocating a volume, no free volumes matching the specified volume were found.

Suggested Action: Check the volume ID and retry command, or add more free volumes with matching criteria.

NOVOLUMES, no volumes match selection criteria

Explanation: When allocating a volume, no volumes match the specified selection criteria.

Suggested Action: Check the selection criteria. Specifically, check the relevant volume pool. If free volumes are in a volume pool, the pool name must be specified in the allocation request, or you must be a default user defined in the pool. You can re-enter the command specifying the volume pool as long as you are an authorized user. Also check that newly-created volumes are in the FREE state rather than the UNINITIALIZED state.

OBJECTEXISTS, specified object already exists

Explanation: The specified object already exists and cannot be created.

Suggested Action: Use a set command to modify the object, or create a new object with a different name.

OBJNOTEXIST, referenced object !AZ does not exist

Explanation: When attempting to allocate a drive or volume, you specified a selection object that does not exist.

Suggested Action: Check spelling of selection criteria objects and retry, or create the object in the database.

OBJREFZERO, dereferenced object with zero count

Explanation: The MDMS server software detected an internal inconsistency. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

PARTIALSUCCESS, some volumes in range were not processed

Explanation: On a command using a volume range, some of the volumes in the range were not processed.

Suggested Action: Verify the state of all objects in the range, and issue corrective commands if necessary.

POLUNDEFINED, referenced pool !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a pool name that does not exist.

Suggested Action: Check spelling of the pool name and retry, or create the pool object in the database.

POOLEXISTS, specified pool already exists

Explanation: The specified pool already exists and cannot be created.

Suggested Action: Use a set command to modify the pool, or create a new pool with a different name.

PROFERROR, profile specification error

Explanation: You specified an invalid user profile for the environment. Verify that the user name specified (default is ABS) exists on the specified node or cluster.

Suggested Action: Re-enter with a valid combination of node or cluster name and user name.

QUEUED, operation is queued for processing

Explanation: The asynchronous request you entered has been queued for processing.

Suggested Action: You can check on the state of the request by issuing a show requests command.

RDFERROR, error allocating or deallocating RDF device

Explanation: During an allocation or deallocation of a drive using RDF, the RDF software returned an error.

Suggested Action: The error following this error is the RDF error return.

REQUESTID, request ID is !@UL

Explanation: The number is the request ID for the command just queued.

Suggested Action: None.

RESUNDEFINED, referenced restore(s) !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a restore name that does not exist. One or more of the specified restores may be undefined.

Suggested Action: Check spelling of the restore names and retry, or create the restore objects in the database.

SAVUNDEFINED, referenced save(s) !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a save name that does not exist. One or more of the specified saves may be undefined.

Suggested Action: Check spelling of the save names and retry, or create the save objects in the database.

SCHEDCREATEERR, failed to create a scheduling job

Explanation: MDMS failed to create a scheduling job.

Suggested Action: Report the incident to VMS Software.

SCHEDDELETEERR, failed to delete a scheduling job

Explanation: MDMS failed to delete a scheduling job.

Suggested Action: Report the incident to VMS Software.

SCHEDDISCONNECT, scheduler disconnected from mailbox

Explanation: The scheduler was disconnected from a mailbox.

Suggested Action: Report the incident to VMS Software.

SCHEDDUPJOB, duplicate scheduler job found

Explanation: MDMS found a duplicate scheduling job.

Suggested Action: Report the incident to VMS Software.

SCHEDEXTSTATUS, external schedule job exited with bad status

Explanation: An external schedule job exited with bad status

Suggested Action: Report the incident to VMS Software.

SCHEDLOOPERR, schedule thread terminating with fatal error, restarting

Explanation: The MDMS internal schedule thread encountered an error and terminated. The thread is restarted.

Suggested Action: Report the problem to VMS Software.

SCHEDMODIFYERR, failed to modify a scheduling job

Explanation: MDMS failed to modify a scheduling job.

Suggested Action: Report the incident to VMS Software.

SCHEDNOJOBCOMPLETE, no job complete time was returned from a scheduled job

Explanation: No job complete time was returned from a scheduled job.

Suggested Action: Report the incident to VMS Software.

SCHEDNOJOBEXISTS, no job exists was returned from a scheduled job

Explanation: No job exists was returned from a scheduled job.

Suggested Action: Report the incident to VMS Software.

SCHEDNOJOBNUM, no job number was returned from a scheduled job

Explanation: No job number was returned from a scheduled job.

Suggested Action: Report the incident to VMS Software.

SCHEDNOJOBSTART, no job start time was returned from a scheduled job

Explanation: No job start time was returned from a scheduled job.

Suggested Action: Report the incident to VMS Software.

SCHEDNOJOBSTATUS, no job status was returned from a scheduled job

Explanation: No job status was returned from a scheduled job.

Suggested Action: Report the incident to VMS Software.

SCHEDNOSUCHJOB, failed to find a scheduling job

Explanation: MDMS failed to find a scheduling job.

Suggested Action: Report the incident to VMS Software.

SCHEDSHOWERR, failed to show a scheduling job

Explanation: MDMS failed to show a scheduling job.

Suggested Action: Report the incident to VMS Software.

SCHEDSYSTEMERR, failed to access the internal scheduler queue

Explanation: An MDMS call to a system service failed in the scheduler functions.

Suggested Action: Report the incident to VMS Software.

SCHEDULECONFL, schedule qualifier and novolume qualifier are incompatible

Explanation: The **/SCHEDULE** and **/NOVOLUME** qualifiers are incompatible for this command.

Suggested Action: Use the **/SCHEDULE** and **/VOLSET** qualifiers for this command.

SCHEDVOLCONFL, schedule qualifier and volume parameter are incompatible

Explanation: The **/SCHEDULE** qualifier and the volume parameter are incompatible for this command.

Suggested Action: Use the **/SCHEDULE** qualifier and leave the volume parameter blank for this command.

SCHEDULECONFL, schedule qualifier and novolume qualifier are incompatible

Explanation: The **/SCHEDULE** and **/NOVOLUME** qualifiers are incompatible for this command.

Suggested Action: Use the **/SCHEDULE** and **/VOLSET** qualifiers for this command.

SCHEDVOLCONFL, schedule qualifier and volume parameter are incompatible

Explanation: The **/SCHEDULE** qualifier and the volume parameter are incompatible for this command.

Suggested Action: Use the **/SCHEDULE** qualifier and leave the volume parameter blank for this command.

SCHUNDEFINED, referenced schedule(s) !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a schedule name that does not exist. One or more of the specified schedules may be undefined.

Suggested Action: Check spelling of the schedule names and retry, or create the schedule objects in the database.

SELUNDEFINED - referenced selection(s) !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a selection name that does not exist. One or more of the specified selections may be undefined.

Suggested Action: Check spelling of the selection names and retry, or create the selection objects in the database.

SETLOCALEFAIL, an error occurred when accessing locale information

Explanation: When executing the SETLOCALE function an error occurred.

Suggested Action: A user should not see this error.

SETPROTECTED, protected field(s) set, verify consistency

Explanation: You have directly set a protected field with this command. Normally, these fields are maintained by MDMS. This has the potential to make the database inconsistent and cause other operations to fail.

Suggested Action: Execute an **MDMS SHOW/FULL** command on the object or objects you have just modified and verify that your modifications leave the objects in a consistent state.

SLSDBINUSE, SLS\$DB network object in use

Explanation: The MDMS server could not be started because it could not declare the network task SLS\$DB.

The network task SLS\$DB is already in use.

Suggested Action: Check the server's logfile for more information.

Check the logical MDMS\$SUPPORT_PRE_V3 in the system table. If this is TRUE and the SLS\$TAPMGRDB process is running, the server cannot be started. Shut down the SLS\$TAPMGRDB process by shutting down SLS. Restart MDMSV3.0 server and then restart SLS.

SNDMAILFAIL, send mail failed, see log file for more explanation

Explanation: While sending mail during the scheduled activities, a call to the mail utility failed.

Suggested Action: Check the log file for the failure code from the mail utility.

SOMESUCCESS, some objects in list were not processed

Explanation: The request was partially successful, but some of the objects were not processed as shown in the extended status.

Suggested Action: Examine the extended status, and retry command as needed.

SPABUFOVR, Space qualifier buffer overflow

Explanation: A character string that is too long was found in an internal line buffer while setting the space qualifier.**Suggested Action:** Ensure that the number of characters are less than or equal to 8 and Re-enter the command.

SPAWNCMDBUFOVR, spawn command buffer overflow

Explanation: During the mount of a volume, the spawned mount command was too long for the buffer. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis.

SVRBUGCHECK, internal inconsistency in SERVER

Explanation: The MDMS server software (MDMS\$SERVER.EXE) detected an inconsistency. This is an internal error.

Suggested Action: Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis. Restart the server.

SVRDISCON, server disconnected

Explanation: The server disconnected from the request because of a server problem or a network problem.

Suggested Action: Check the server's logfile and file SYS\$MANAGER:MDMS\$SERVER.LOG for more information. Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

SVREXIT, server exited

Explanation: Server exited. Check the server logfile for more information.

Suggested Action: Depends on information in the logfile.

SVRLOGERR, server logged error

Explanation: The server failed to execute the request. Additional information is in the server's logfile.

Suggested Action: Depends on information in the logfile.

SVRRUN, server already running

Explanation: The MDMS server is already running.

Suggested Action: Use the MDMS\$SHUTDOWN procedure with parameter RESTART to restart the server.

SVRSTART, Server !AZ!UL.!UL-!UL started

Explanation: The server has started up identifying its version and build number.

Suggested Action: None.

SVRSTARTSTRING, Server !AZ started

Explanation: The server has started up identifying its version and build number.

Suggested Action: None.

SVRTERM, Server terminated abnormally

Explanation: The MDMS server was shut down. This could be caused by a normal user shutdown or it could be caused by an internal error.

Suggested Action: Check the server's logfile for more information. If the logfile indicates an error has caused the server to shut down then provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

SVRUNEXP, unexpected error in SERVER !AZ line !UL

Explanation: The server software detected an internal inconsistency.

Suggested Action: Provide copies of the MDMS command issued, the database files and the server's logfile for further analysis.

TCPIPLISEXIT, TCP/IP listener exited

Explanation: The TCP/IP listener has exited due to an internal error condition or because the user has disabled the TCPIP transport for this node. The TCP/IP listener is the server's routine to receive requests via TCP/IP.

Suggested Action: The TCP/IP listener should be automatically restarted unless the TCPIP transport has been disabled for this node. Provide copies of the MDMS command issued, the database files, and the server's logfile for further analysis if the transport has not been disabled by the user.

TCPIPLISRUN, listening on TCP/IP node !AZ port !AZ

Explanation: The server has successfully started a TCP/IP listener. Requests can now be sent to the server via TCP/IP.

Suggested Action: None.

TOOLARGE, entry is too large

Explanation: Either entries cannot be added to a list of an MDMS object or existing entries cannot be renamed because the maximum list size would be exceeded.

Suggested Action: Remove other elements from list and try again.

TOOMANY, too many objects generated

Explanation: You attempted to perform an operation that generated too many objects.

Suggested Action: There is a limit of 1000 objects that may be specified in any volume range, slot range or space range. Re-enter command with a valid range.

TOOMANYSELECTS, too many selections for a field, use only one

Explanation: More than one selection was specified for a particular field.

Suggested Action: Specify only one field to select on.

TOOMANYSORTS, too many sort qualifiers, use only one

Explanation: When specify more than one field to sort on.

Suggested Action: Specify only one field to sort on.

UNDEFINEDREFS, success, but object references undefined objects

Explanation: The command was successful, but the object being created or modified has references to undefined objects. Subsequent messages indicate which objects are undefined.

Suggested Action: This allows objects to be created in any order, but some operations may not succeed until the objects are defined. Verify/correct the spelling of the undefined objects or create the objects if needed.

UNKVOLENT, unknown volume !AZ entered in jukebox !AZ

Explanation: A volume unknown to MDMS has been entered into a jukebox.

Suggested Action: Use the **MDMS INVENTORY** command to make the volume known to MDMS or use a jukebox utility program (CARTRIDGE or MRU) to eject the volume from the jukebox.

UNSUPPORTED, unsupported function

Explanation: You attempted to perform an unsupported function.

Suggested Action: None.

UNSUPPORTED1, unsupported function !AZ

Explanation: You attempted to perform an unsupported function.

Suggested Action: None.

UNSUPRECVER, unsupported version for record !AZ in database !AZ

Explanation: The server has detected unsupported records in a database file. These records will be ignored.

Suggested Action: Consult the documentation about possible conversion procedures provided for this version of MDMS.

USERNOTAUTH, user is not authorized for volume pool

Explanation: When allocating a volume, you specified a pool for which you are not authorized.

Suggested Action: Specify a pool for which you are authorized, or add your name to the list of authorized users for the pool. Make sure the authorized user includes the node name or group name in the pool object.

VISIONCONFL, vision option and volume parameter are incompatible

Explanation: You attempted to create volumes with the vision option and the volume parameter. This is not supported.

Suggested Action: The vision option is used to create volumes with the volume identifiers read by the vision system on a jukebox. Re-enter the command with either the vision option (specifying jukebox and slot range), or with volume identifiers, but not both.

VOLALRALLOC, specified volume is already allocated

Explanation: You attempted to allocate a volume that is already allocated.

Suggested Action: Use another volume.

VOLALRINIT, volume is already initialized and contains data

Explanation: When initializing a volume, MDMS detected that the volume is already initialized and contains data.

Suggested Action: If you are sure you still want to initialize the volume, re-enter the command with the overwrite option.

VOLIDICM, volume ID code missing

Explanation: The volume ID is missing in a request.

Suggested Action: Provide volume ID and retry request.

VOLINDRV, volume is currently in a drive

Explanation: When allocating a volume, the volume is either moving or in a drive, and nopreferred was specified.

Suggested Action: Wait for the volume to be moved or unloaded, or use the preferred option.

VOLINJUKE, volume is in a jukebox

Explanation: You attempted load a volume that is currently in a jukebox into a drive that is not in the jukebox.

Suggested Action: Load the volume into a drive within the current jukebox, or check the jukebox name for the drive.

VOLINSET, volume is already bound to a volume set

Explanation: You cannot bind this volume because it is already in a volume set and is not the first volume in the set.

Suggested Action: Use another volume, or specify the first volume in the volume set.

VOLLOST, volume location is unknown

Explanation: The volume's location is unknown.

Suggested Action: Check if the volume's placement is in a magazine, and if so if the magazine is defined. If not, create the magazine. Also check the magazine's placement.

VOLMOVE, volume cannot be loaded but can be moved to jukebox or drive

Explanation: The volume is not currently in a placement where it can be loaded, but can be moved there.

Suggested Action: Move the volume to the drive, or use the automatic move option on the load and retry.

VOLMOVING, volume is currently being moved

Explanation: In a move, load or unload command, the specified volume is already being moved.

Suggested Action: Wait for volume to come to a stable placement and retry. If the volume is stuck in the moving placement, check for an outstanding request and cancel it. If all else fails, manually change volume state.

VOLNOTALLOC, specified volume is not allocated

Explanation: You attempted to bind or deallocate a volume that is not allocated.

Suggested Action: None for deallocate. For bind, allocate the volume and then bind it to the set, or use another volume.

VOLNOTBOUND, volume is not bound to a volume set

Explanation: You attempted to unbind a volume that is not in a volume set.

Suggested Action: None.

VOLNOTINACS, one or more volumes are not in this ACS

Explanation: One or more volumes for the command are not in this ACS.

Suggested Action: Verify that all volumes are in the same ACS and that the ACS id is correct.

VOLNOTINJUKE, volume is not in a jukebox

Explanation: When loading a volume into a drive, the volume is not in a jukebox.

Suggested Action: Use the move option and retry the load. This will issue OPCOM messages to move the volume into the jukebox.

VOLNOTINPOOL, loaded volume is not in the specified pool

Explanation: During a scratch load of a volume in a drive, the volume loaded was not in the requested pool.

Suggested Action: Load another volume that is in the requested pool. A recommended volume is printed in the OPCOM message. Note that if no pool was specified, the volume must have no pool defined.

VOLNOTLOADED, the volume is not loaded in a drive

Explanation: On an unload request, the volume is not recorded as loaded in a drive.

Suggested Action: If the volume is not in a drive, none. If it is, issue an unload drive command to unload it.

VOLONOTDRV, volume is currently in another drive

Explanation: When loading a volume, the volume was found in another drive.

Suggested Action: Wait for the volume to be unloaded, or unload the volume and retry.

VOLSALLOC, !AZ volumes were successfully allocated

Explanation: When attempting to allocate multiple volumes using the quantity option, some but not all of the requested quantity of volumes were allocated.

Suggested Action: See accompanying message as to why not all volumes were allocated.

VOLSDRIVES, one or more of the volumes are in drives or are moving

Explanation: One or more of the volumes in the move request are in drives and cannot be moved. A show volume /brief will identify which volumes are in drives.

Suggested Action: Unload the volumes in drives and retry, or retry without specifying the volumes in drives.

VOLUMEEXISTS, specified volume(s) already exist

Explanation: The specified volume or volumes already exist and cannot be created.

Suggested Action: Use a **SET** command to modify the volumes, or create new volumes with different names.

VOLUNDEFINED, referenced volume !AZ undefined

Explanation: When creating or modifying a valid object, the object's record contains a reference to a volume ID that does not exist.

Suggested Action: Check spelling of the volume ID and retry, or create the volume object in the database.

VOLWRTLCK, volume loaded with hardware write-lock

Explanation: The requested volume was loaded in a drive, but is hardware write-locked when write access was requested.

Suggested Action: If you need to write to the volume, unload it, physically enable it for write, and reload it.

WRONGLABEL, initializing volume !AZ as !AZ is disallowed

Explanation: The label of the volume loaded in the drive for initialization does not match the requested volume label and there is data on the volume. Alternatively, initializing the volume with the requested label causes duplicate volumes in the same jukebox or location.

Suggested Action: If you wish to overwrite the volume label, re-issue the command with the **/OVERWRITE** qualifier. If there are duplicate volumes in the same location or jukebox, you need to move the other volume from the jukebox or location before retrying.

WRONGVOLUME, wrong volume label or unlabelled volume was loaded

Explanation: On an **MDMS LOAD VOLUME** command, MDMS loaded a volume with the wrong volume label or a blank volume label into the drive.

Suggested Action: Check the volume, and optionally perform an initialization of the volume and retry. If this message is displayed in an OPCOM message, you will need another free drive to perform the initialization. The volume has been unloaded.

Chapter 7. ABS Messages

ABS_AOE_INSTANCE_MATCH_FAIL, AOE instance record does not match input

Explanation: ABS internal error. The specified AOE instance object does not match with the corresponding instance object in the catalog. The matching can be with respect to the Common UID, Saveset UID, Date Archived, Archive_Expiration_Date, Object_Revision_Date, Owner, or the File_Section_Number object.

Suggested Action: Contact VMS Software for support.

ABS_AOE_NOT_FOUND, an archive object entry was not found in the catalog

Explanation: Either an archive object entry file is not found in the catalog or a particular record is not found in the AOE file.

Suggested Action: Use a wildcard specification in the catalog reporting facility to verify the archive object name. Ensure that you have specified the correct data object name. If using the wildcard specification does not provide the correct data object, then it implies that the valid save operations of the object have not been performed.

ABS_AOE_SHOW_CONTEXT, AOE Show Context is not NULL

Explanation: ABS internal error. The error has occurred in the AOE_SHOW_CONTEXT.

Suggested Action: Contact VMS Software for support.

ABS_AOE_VALIDATE_FAIL, common AOE information specified does not match the catalog

Explanation: ABS internal error. The specified AOE object information does not match the corresponding object in the catalog. The matching can be with respect to the Device name, Object name, Object version, Node name, Object_Creation_Date, or Object_Type_Name object.

Suggested Action: Contact VMS Software for support.

ABS_CATLG_DB_NOT_FOUND, catalog not found in the Catalog database

Explanation: While ABS is able to open the ABS_CATALOG_OBJECTS.DAT file, it cannot find the specific catalog name in the file.

Suggested Action: Follow these steps to add a catalog to the catalog database or provide the necessary information (as shown) when prompted by the Catalog Create process:

\$ CATLG_OBJECT

- Would you like to CREATE or SHOW the catalog object ([CREATE]/SHOW): Create
- Enter catalog name: Example_Catalog
- Enter catalog type ([BRIEF], SLS):
(Select type as BRIEF if the catalog is for pab native catalogs)
- Enter catalog owner [CURRENT_PROCESS]:
- Does this catalog use staging (Yes/[No]): YES
(Specify YES if you want to use the staging catalog option)

ABS_CONVCAT, catalog needs to be converted - long filename(s) not entered

Explanation: Catalogs with long names are not converted.

Suggested Action: Convert the catalogs with long names to the normal format by executing the following catalog conversion procedure:

ABS\$ROOT : [SYSTEM]ABS\$CONVERT_CATALOG.COM

ABS_EXTEND_SUBSTATE_FINISHED, archive Extend substate finished

Explanation: ABS internal error. One step in the archive extend operation has completed. It implies that one of the several interactions with the agent, which is needed to extend the archive has completed.

Suggested Action: Contact VMS Software for support.

ABS_FAILURE, a failure occurred

Explanation: ABS internal error. A failure has occurred.

Suggested Action: Contact VMS Software for support.

ABS_INTERNAL_ERROR, unexpected error occurred in an ABS module

Explanation: ABS internal error. An internal error has occurred in one of the ABS modules.

Suggested Action: Contact VMS Software for support.

ABS_INV_CAT_AFS_TYPE, volume set catalog type not supported for disk archive

Explanation: Disk type of archive object cannot have Volume_Set type of catalog specified.

Suggested Action: Remove the Volume_Set type of catalog specification.

ABS_INVLD_AGENT_ROOT, agent file system root is invalid

Explanation: The backup agent file system root is invalid (Agent_Filesystem_Root is too long).

Suggested Action: Specify a shorter Agent_Filesystem_Root.

ABS_INVLD_CATALOG_TYPE, invalid catalog type

Explanation: ABS internal error. The catalog type does not match any of the ABS support types of catalogs.

Suggested Action: Contact VMS Software for support.

ABS_INVLD_DATE_FORMAT, invalid data format

Explanation: ABS internal error. ABS is unable to format the binary date to an ASCII date format.

Suggested Action: Contact VMS Software for support.

ABS_INVLD_DEVICE_NAME, invalid device name

Explanation: ABS internal error. An invalid device name is specified.

Suggested Action: Contact VMS Software for support.

ABS_INVLD_FILE_NAME, invalid file name

Explanation: An invalid Archive file system file name is specified.

Suggested Action: Check the Archive file system's file name. If it contains invalid characters, contact VMS Software for support.

ABS_INVLD_INCLUDE_SPEC, invalid include file spec

Explanation: When a version number greater than four digits is specified in the include specification, then the ABS Save, Restore, and Lookup requests cannot manage the include specification. In addition, when reporting the above-mentioned error, ABS replaces the device component of the file specification with the current default device.

Suggested Action: Specify a valid include specification.

ABS_INVLD_NEWVOL_PROTOCOL, new volume protocol for agent is invalid

Explanation: An invalid new volume protocol for the backup agent is specified. This results in incorrect backup agent information.

Suggested Action: If you have modified the template settings, restore the original template settings from the ABS distribution kit. If you have not modified the template information but are still getting this error, then contact VMS Software for support.

ABS_INVLD_NODE_NAME, invalid node name

Explanation: An invalid node name is specified.

Suggested Action: Specify a valid node name.

ABS_INVLD_OBJECT_NAME, invalid object name

Explanation: ABS internal error. An invalid object name is specified.

Suggested Action: Contact VMS Software for support.

ABS_INVLD_OBJECT_TYPE, invalid storage object type

Explanation: ABS internal error. An invalid storage object type is specified (such as a UID mismatch).

Suggested Action: Contact VMS Software for support.

ABS_INVLD_OBJECT_VERSION, must specify object name with object version

Explanation: An object version is specified without a valid object name.

Suggested Action: Specify a valid object name when you specify the object version.

ABS_INVLD_PATH_NAME, invalid path name

Explanation: An invalid path name is specified.

Suggested Action: Specify a valid path name.

ABS_INVLD_RETENTION_CRITERIA, invalid retention criteria

Explanation: An invalid retention criteria structure is specified.

Suggested Action: Specify a valid retention criteria parameter.

ABS_INVLD_SELECT_CRITERIA, invalid selection criteria structure

Explanation: ABS internal error. An invalid selection criteria structure is specified.

Suggested Action: Contact VMS Software for support.

ABS_INVLD_TEMPLATE_STATE, invalid state specified in tag template

Explanation: An invalid state is specified in the tag template. This results in incorrect backup agent information.

Suggested Action: If you have modified the template settings, restore the original template settings from the ABS distribution kit. If you have not modified the template information but still getting this error, then contact VMS Software for support.

ABS_NO_AOE_SHOW_CONTEXT, ABS Internal error - AOE Show Context is NUL

Explanation: ABS internal error. AOE show context is NULL

Suggested Action: Contact VMS Software for support.

ABS_NO_CATALOG, a catalog must be specified

Explanation: A valid catalog name is not specified.

Suggested Action: Specify a valid catalog name.

ABS_NO_MORE_AOE_ENTRIES, no more entries match AOE selection criteria

Explanation: There are no more AOE entries that are matching the selection criteria in the catalog.

Suggested Action: It is just an ABS internal informational message.

ABS_NO_WILD_MATCH, wildcard match for the object was not found

Explanation: There are no more entries that are matching the AOE selection criteria.

Suggested Action: Do not issue another ABS_ShowObjectEntry.

ABS_NOMEM, ABS was not able to allocate virtual memory

Explanation: ABS could not allocate the virtual memory.

Suggested Action: Check the page file quota for the account executing the unsuccessful job. Increase the page file quota. If the problem persists, contact VMS Software for support.

ABS_PIPEDELETEERR, error deleting pipe

Explanation: ABS internal error. There is an error when deleting the pipe.

Suggested Action: Contact VMS Software for support.

ABS_SCRATCH_RVN_MISMATCH, media Set RVN does not match scratch volume

Explanation: The name of the RVN in the media set is not matching the given scratch volume for the data to be written.

Suggested Action: Set up the media set properly for the volume on the drive by appending the scratch volume as needed.

ABS_SPECIFY_DEVICE, device name must be specified

Explanation: The device name is not specified.

Suggested Action: Specify a physical disk, System_Logical_Name, or Requestor_Logical_Name.

ABS_SPECIFY_OBJECT, object name must be specified

Explanation: The object name is not specified.

Suggested Action: Specify the object name that must also be there in the catalog.

ABS_STRING_OVERFLOW, string copy overflowed output buffer

Explanation: ABS internal error. The output buffer is overflowed in the string operations.

Suggested Action: Contact VMS Software for support.

ABS_SUCCESS, normal successful completion

Explanation: The operation was completed successfully.

Suggested Action: None.

ABS_TLE_NOT_FOUND, a transaction log entry was not found in the catalog

Explanation: The transaction log entry is not found in the catalog. Based on some scenarios, this can either be an informative message or an ABS internal error.

Suggested Action: If it is an ABS internal error, contact VMS Software for support.