Section 1 - User Commands

DeepSpace Documentation

Exported on 05/05/2020
DRAFT

Table of Contents

This section provides individual man pages for each of the the following DeepSpace user commands:

1 ds(1)

1.1 Name

`ds` — DeepSpace command-line interface.

1.2 Synopsis

`ds`

On this page:

1.3 Description

Many features of DeepSpace are accessible from a command-line interface. Online help is available.

1.4 See Also

dsedit(1)(see page 12)

DeepSpace Catalog Data Services Operations Guide[1]

1 https://deepspace.atlassian.net/wiki/spaces/DeepSpaceDoc/pages/247922689/DeepSpace+Catalog+Data+Services+Operations+Guide

2 dsedit(1)

2.1 Name

`dsedit` — modify the catalog entry of a volume.

2.2 Synopsis

`dsedit [`*`keywords`*`] `*`vid`*

On this page:

2.3 Description

`dsedit` modifies the catalog entry of volume *vid* to reflect the given keyword assignments. A volume may only be edited by its owner.

2.4 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. Keywords listed are optional with no default specified. See Keywords(see page 13) for a list of all valid keywords.
`vid`	Volume ID. A unique identifier assigned to each volume when it is submitted. It can be up to twelve characters long.

2.5 Keywords

Keyword	Valid Value
app	= Application prefix. This prefix is used with file reports to limit selections to those with the specified `app`= value. app may be up to 12 characters long.
`capacity`	= Volume capacity in Mbytes. Specifies the volume capacity for cartridge media types.
conv	= Record conversion specification. This value controls the conversion of records to and from tape. `conv` may be one of the following: • text for text records. • etext for EBCDIC text records • data for fixed length. • ASCII binary data records • edata for fixed length. • EBCDIC data records. Under normal circumstances, this field should not be edited. If the file resides on a labeled tape and the database value for record conversion does not match the value recorded in the tape label, all accesses to the file will be aborted.
`dispose`	= Volume disposition. This value controls when volumes leave the volumeset and what happens when they do. The `dispose` keyword may be set to erase and/or retain. If erase is specified, volumes that leave the volumeset (when it is truncated or scratched) are erased before they can be reallocated. If retain is specified the volumeset is never truncated. (Truncation occurs when the first file on a volumeset is overwritten and the new file is not large enough to span all the volumes currently in the volumeset.) To select both, specify `dispose=erase&retain`.
ecnt	=Error count. Set error count as specified.
`finger`	= File fingerprint. Set fingerprint field as specified.
`flocation`	= Free volume location. When a volumeset is truncated or scratched, the newly unattached volumes will return to the location assigned to `flocation`. The value given must be defined as a DeepSpace storage site (the `dsreport --sites` command will produce a list of currently defined storage sites).

Keyword	Valid Value
`format`	= Label Format. `format` may be set to one of the following: • ANSI • IBM • IBMU • TAR • CPIO • RAW

`ftemplate`	= File Name template. The filename template is used to dynamically construct names for files written to the volumeset. The name constructed by the template is limited to seventeen (17) characters. The template consists of constant text and substitution patterns. Substitution patterns have the general form:
	`@sublen@`
	In this pattern,
	
	• @ symbol delimits the beginning and end of the substitution specification.
	• *`sub`* is a character indicating what to substitute.
	• *`len`* gives the length in characters of the substituted value. In principle can be any number between 1 and 17, however, for some values of sub only a particular value of len is reasonable.
	
	The following values are recognized:
	
	• numeric year
	• numeric month (Jan = 1)
	• month name
	• day of month
	• Julian day
	• day of week (Sun = 1)
	• hour (24 hour clock)
	• minute
	• second
	• value assigned fid keyword on the `dsvwrite` command
	• base name of file assigned to the if keyword on the `dsvwrite` command
	• file generation
	• file version
	
	Numeric values are truncated on the left. Character values are truncated on the right. If a numeric value is shorter then *`len`* it is padded on the left with zeros: for example,
	`January 25 1987: ftemplate=GV`
	
	results in a file name of:
	
	`25Jan89G0000V01`

Keyword	Valid Value
`ftrack`	= File Tracking Flag. A value of yes causes the catalog to maintain a record for every file written to the volumeset. A value of no disables file cataloging. Maintaining file catalogs makes it possible to generate a volumeset table of contents without having to mount a tape. It also improves the efficiency of accessing files on multi-volume volumesets.
`group`	= File group. The group (from `/etc/group`) to which the file belongs. It can be up to 12 characters long.
`initialize`	= Volumeset Initialization required. A value of yes indicates the volumeset requires initialization. The first time each volume is mounted, DeepSpace will initialize it. A value of no indicates the volumeset contains existing data. The first time each volume is mounted DeepSpace checks to make sure the Volume Serial Number on the volume label matches the Volume Serial Number in the catalog. The default is yes.
`offset`	= Record offset. The number of bytes reserved at the beginning of each block for additional information. Offset only affects the ANSI tape format. Under normal circumstances, this field should not be edited.
`passwd`	= Volume Access Password. An optional password which, if specified, must be given before the volumeset can be read or written: for example, passwd=undertow
`pool`	= Pool membership. Every volume must belong to a pool. By default, volumes belong to the user's private pool. Volumes may only be assigned to pools owned by the current effective user ID. The full specification of pool is userid/pool. If userid is omitted, the current effective user ID is assumed. Only "operator" users may create pools for IDs other than their own.

Keyword	Valid Value
`rformat`	= Record format. The new record format for the file. `rformat` takes the following form: *`fmt`*`:`*`blen`*`:`*`rlen`* In this syntax, *`fmt`* is one of the following: • f - fixed length records. • fb - fixed length, blocked records. • v - variable length records. • vb - variable length, blocked records. • vs - variable length, spanned records. • vbs - variable length, blocked, spanned records. • u - unformatted data. *`blen`* is the block length in bytes. f*`len`* is the record length in bytes. For example: `rformat=fb:800:80`
`scratch`	= Scratch status. Sets volume scratch status (scratches entire volumeset if volume is the first member of the volumeset).
`user`	=File owner. The user ID (from `/etc/passwd`) to which the file belongs. This can be up to 12 characters long.
`vaccess`	= Volume Access Byte. The character assigned to `vaccess` is written as byte 11 (Volume Accessibility) in the VOL1 label. DeepSpace attaches no particular significance to the Volume Access byte; control is provided for export of volumes to sites that require certain values. Applies to IBM and ANSI formats only. The default is 0x00.

Keyword	Valid Value
`vexpire`	= Volumeset Expiration Date, which can be one of the following: • I - infinite (never expires). • S - scratch (immediately expired). • RN - expires N days after creation. • AN - expires if not accessed in N days. • L - expires when all files on the volumeset have expired. • O*rotsched* - follows the rotation schedule *rotsched* (see XREF dsrcreate(1)). • *Xccyymmdd* or *Xmm/dd/yy* - expires on given date. • GN - expire when there are N newer generations. Note that upon expiration, the volumeset is not disbanded until it is scratched. The `dsvscratch` program disbands volumesets. For example, `vexpire=R30` The default is S.
`vmode`	= Volumeset Permission Mask. Three octal digits controlling owner, group and others (respectively) permission to the volumeset (similar to the UNIX file mode). Bit 4 controls read access. Bit 2 controls write access. Bit 1 permits viewing of the volume's catalog entry. For example: `vmode=744` The default is 700. The `vmode` keyword is only applicable to the first volume in a volumeset.
`vsn`	= Volume Serial Number. The volume serial number (VSN) associated with the volume. The VSN is written in the VOL1 label for IBM and ANSI tape formats. Even unlabeled volumes (IBMU, TAR, CPIO and RAW) must be given a VSN. In general DeepSpace does not require each volume to have a unique VSN. This can be up to six characters long.
`slocation`	= Scheduled Volume Location. If a volume's current location (`clocation`) does not equal its scheduled location. The volume shows up on the Volume maintenance report as waiting to move from `clocation` to `slocation`. The value given must be defined as an DeepSpace storage site. (The `dseport sites` command produces a list of currently defined storage sites.)
`type`	= Volume Media Type. Media Type is an arbitrary name assigned during DeepSpace configuration to describe Volume formats supported by devices under DeepSpace control. For reel media type devices, this is typically the tape density (i.e. 1600, 6250, etc.). For cartridge devices, it is typically the cartridge standard (i.e. LTO, 9840C, etc.). The given value must be a defined Media Type. The `dsreport --types` command produces a list of defined Media Types.

Keyword	Valid Value
The following keywords may only be used by DeepSpace operators:	
`clean`	= Volume cleaning count. The number of times the volume has been mounted since it was last cleaned. This value is normally maintained by DeepSpace.
`clocation`	= Current Volume Location. The value given must be defined as a DeepSpace storage site. (The `dsreport --sites` command produces a list of currently defined storage sites.)
`ctype`	= Current Media Type. For initialized volumes, this field contains the Media Type DeepSpace has verified is the correct media type for the volume. This field is normally maintained by DeepSpace.
`rack`	=Rack Number. The rack number identifying the volume's storage slot. The `rack` keyword is provided here so operators can submit volumes in a single step. It can be up to twelve characters long. If volume acceptance is required, the rack number is normally assigned when the operator accepts the volume.
`vid`	= Volume ID. A unique volume identifier. For sites that maintain unique Volume Serial Numbers, `vid` is normally identical to `vsn`. By default, DeepSpace, assigns each volume a unique volume ID by combining a four character word with a three digit number: for example, root-496. It can be up to twelve characters long.
`maintenance`	= Scheduled Maintenance. Maintenance may be set to one or more of the folllowing: • move - indicates the volume is awaiting movement. • age - causes the volume to wait *N* days after it is scratched before becoming free. (*N* is defined by the pool the tape belongs to.) • erase - schedules the volume for erasure. • clean - schedules the volume for cleaning. Multiple items are selected by separating them with the `&` (ampersand) character: for example, `maint="erase&clean"` Normally the maintenance value is maintained by DeepSpace.
`remove`	= Volume Usage Count. The number of times the volume has been mounted since it entered the catalog. This value is normally maintained by DeepSpace.

Keyword	Valid Value
`status`	= Volume submission status. Status may be assigned the following integer values. • 1 - submitted awaiting acceptance. • 2 - accepted awaiting identification. • 3 - fully submitted. • 4 - retrieved awaiting return. Submission status is normally maintained by DeepSpace.

2.6 See Also

dspsubmit(1)(see page 39)
dsvsubmit(1)(see page 92)
dsvaccess(1)(see page 61)

3 dsfedit(1)

3.1 Name

`dsfedit` — modify the catalog entry of a file.

3.2 Synopsis

```
dsfedit keywords fname
dsfedit vol=volset keywords fname
dsfedit vol=volset fseq=fsn keywords
```

On this page:

3.3 Description

`dsfedit` modifies the catalog entry of the specified file. A file may be specified by *`fname`* only if *`fname`* is unique in the library. A file must be specified with *`fname`* and the `vol=` keyword if *`fname`* occurs on other volumesets in the library. If the file is unnamed, or if *`fname`* occurs more than once on the specified volumeset, the file must be specified with the `vol=` and `fseq=` keywords.

A file may only be edited by its owner. Files on volumesets that are in use cannot be edited.

3.4 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. See Keywords(see page 21) for a list of all valid keywords.
`fname`	The cataloged name of the file to edit. If *`fname`* is not unique in the library, the `vol=` keyword is required. If *`fname`* is not unique on the volumeset, the `fseq=` keyword is required.

Argument	Valid Value
`vol=`*`volset`*	The cataloged name of the volumeset, assigned by the volumeset owner. Volumeset names take the form `[`*`userid/`*`]`*`vname`*`[:`*`Ggno`*`][:`*`Vvno`*`]`. If *`userid`*is omitted, the effective user ID is assumed. *`vname`* is an arbitrary string up to 12 characters long. *`Ggno`* specifies a volumeset generation number. *`Vvno`* specifies a volumeset version number.
`fseq=`*`fsn`*	The file sequence number. Selects a file to edit by its relative position on the volumeset. fseq=1 selects the first file on the volumeset, fseq=2 selects the second file, and so on. You may not edit this field. Use it for file identification in cases where fname does not exist, or is not unique on the volumeset.

3.5 Keywords

Key word	Valid Value

`fid`	=File ID. The new file ID to associate with the specified file. File IDs take the following form: `fname[:Ggno][:Vvno]`.
	
	In this syntax,
	
	• `fname` is the file name, an arbitrary string up to 17 characters long.
	• `Ggno` is the file generation number.
	• `Vvno` is the file version number. Generation and version numbers are used as subscripts for files with the same `fname`. This is explained below.
	  • gno can be an integer or a signed integer. If gno is an integer, it references the given generation number. If gno is a signed integer, it references the highest existing generation number offset by gno. For example, if the highest generation number for file forecast is 10, :G-1 refers to generation 9 and :G+1 refers to generation 11. If the generation specification is omitted, the highest existing generation is referenced. If there are no existing generations, generation 0 is selected. The highest allowable generation number is 9999.
	  • vno specifies a file version number. Version numbers behave like generation numbers. The highest allowable version number is 99. Under ordinary circumstances, this field should not be edited; if the file resides on a labeled tape, the database value for file ID must match the value recorded on tape in the HDR2 label. If these values do not match, all accesses to the file will be aborted.

Key word	Valid Value
`fexpire`	=File expiration date. The new expiration date for the file, which can be one of the following: • I - infinite (never expires). • S - scratch (always expires). • R*N* - expires *N* days after creation (maximum: 999). • A*N* - expires if not accessed in *N* days. • X*ccyymmdd* or X*mm*/*dd*/*yy* - expires on given date. • G*N* - expire when there are *N* newer generations. File expiration is only considered if volumeset expiration is set to L.
`rformat`	= Record format. The new record format for the file. `rformat` takes the following form: *`fmt`*`:`*`blen`*`:`*`rlen`* In this syntax, *`fmt`* is one of the following: • f - fixed length records. • fb - fixed length, blocked records. • v - variable length records. • vb - variable length, blocked records. • vs - variable length, spanned records. • vbs - variable length, blocked, spanned records. • u - unformatted data. *`blen`* is the block length in bytes. f*`len`* is the record length in bytes. For example: `rformat=fb:800:80`
`offset`	= Record offset. The number of bytes reserved at the beginning of each block for additional information. Offset only effects the ANSI tape format. Under ordinary circumstances, this field should not be edited; if the file resides on a labeled tape, and the database value for file offset does not match the value recorded in the tape label, all accesses to the file will be aborted.

Key word	Valid Value
`conv`	= Record conversion specification. This value controls the conversion of records to and from tape. conv may be one of: text for text records; etext for EBCDIC text records, data for fixed length ASCII or binary data records; edata for fixed length EBCDIC data records. Under ordinary circumstances, this field should not be edited; if the file resides on a labeled tape, and the database value for record conversion does not match the value recorded in the tape label, all accesses to the file will be aborted.
`app`	= Application prefix. This prefix is used with file reports to limit selections to those with the specified `app=` value. `app` may be up to 12 characters long.
`fcomment`	= File comment. A comment to associate with the file. `fcomment` may be up to 40 characters long. If the comment includes spaces, it must be enclosed in quotes.
`fpasswd`	= File access password. A password to associate with the file. `fpasswd` may be up to 14 characters long. Files with passwords will not be accessed unless the password is provided.
`user`	=File owner. The user ID (from `/etc/passwd`) to which the file belongs. `user` may be up to 12 characters long.
`group`	= File group. The group ID (from `/etc/group`) to which the file belongs. `group` may be up to 12 characters long.
`fmode`	= File permission mask. Three octal digits that control access to the file. For example, the following sets the UNIX file permission string to rwxr-r-: `fmode=744`

3.6 Examples

- The following command edits the expiration date. Because the filename `payroll` is not unique in the library, the volumeset on which it resides is specified.
  ```
  dsfedit vol=accounting fexpire=X12/31/95 payroll
  ```
- The following command edits the permission mask of the file to extend read, write, and execute privileges to all users, groups, and others. Because the file does not have a cataloged name, it is specified with the `fseq=` keyword.
  ```
  dsfedit vol=project37 fseq=3 fmode=777
  ```

3.7 See Also

dsreport(1)(see page 46)

dsedit(1)(see page 12)
dsvedit(1)(see page 72)

4 dsfree(1)

4.1 Name

`dsfree` — free a previously reserved device.

4.2 Synopsis

`dsfree [psd]`

On this page:

4.3 Description

`dsfree` frees pseudo devices previously allocated by `dsreserve(1)`. If the time-out period between `dsreserve` and `dsvmount` expires and device reservation is lost, you must issue a `dsfree` command to free that pseudo device name before attempting the request again.

4.4 Options

Argument	Valid Value
psd	Pseudo Device Name. An arbitrary name assigned during device reservation. This is used to distinguish between devices when multiple devices are reserved. If *psd* is not specified, all devices under the user's control are freed.

4.5 See Also

dsstatus(1)(see page 57)

dsreserve(1)(see page 53)

5 dsinq(1)

5.1 Name

`dsinq` — Displays the path names for the special files associated with a pseudo device name. `dsinq` must run from the host running the same version of `ds` as the client.

5.2 Synopsis

`dsinq [-n] [-r] [psd]`

On this page:

5.3 Description

`dsinq` without any options displays the rewind and no-rewind special file names associated with the given pseudo device name. With the `-n` option, `dsinq` displays the no-rewind device name in a format suitable for assignment to a shell variable. Similarly, the `-r` option displays the rewind device name.

5.4 Options

Argument	Valid Value
`-n`	Display the no-rewind device name in a format suitable for assignment to a shell variable.
-r	Display the rewind device name in a format suitable for assignment to a shell variable.

Argument	Valid Value
psd	Pseudo Device Name. An arbitrary name assigned during device reservation. This is used to distinguish between devices when multiple devices are reserved. If a device reservation is not made by ADN, `dsinq` will not display anything until there is actually a mounted volume. The reason for this is the actual device assigned to a request can change up until the point where a requested volume is actually mounted.

5.5 See Also

dsstatus(1)(see page 57)

6 dsmain(1)

6.1 **TBD....**

6.2 Name

`dsmain`— DeepSpace screen interface.

6.3 Synopsis

`dscreate` *`rotsched`*

On this page:

6.4 Description

`dsrcreate` creates a rotation schedule named *`rotsched`*. A rotation schedule is a list of one or more “location-type-count” triples. Volumesets can be assigned rotation schedules....

6.5 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. See Keywords for a list of all valid keywords.
`rotsched`	A list of one or more “location-type-count” triples. Volumesets can be assigned rotation schedules....

6.6 Keywords

Argument	Valid Value
auacc	= A comma-separated list of user IDs to add to the user access list.

6.7 Examples

6.8 Return Values

State	Return Value
Success	The number of bytes in the converted record.
Error	-1 and `terrno` indicates the error condition.

6.9 See Also

Content by label

There is no content with the specified labels

7 dsopmsg(1)

Name

`dsopmsg` — send a message to the DeepSpace operator(s).

7.1 Synopsis

`dsopmsg` *`message`*

On this page:

7.2 Description

`dsopmsg` sends the string message to all currently active DeepSpace operators.

7.3 Options

Argument	Valid Value
`message`	A message to active operators. The message cannot be larger than 65 characters.

7.4 See Also

Section 1 - User Commands(see page 10)

8 dspcreate(1)

Name

`dspcreate` — create a new volume pool.

8.1 Synopsis

`dspcreate [keywords] pool`

On this page:

8.2 Description

`dspcreate` creates a new volume pool. Pools are used to make volumes available to selected users or applications. Any user can create a pool. The creator of a pool specifies what user IDs are permitted to allocate volumes from the pool. By default, only the creator of a pool may allocate pool volumes.

The full specification of pool is userid/pool. If userid is omitted, the current effective user ID is assumed. Only "operator" users may create pools for IDs other than their own. The pool `private` is implicitly created the first time a user submits a volume to DeepSpace. If a user wishes to designate a set of volumes for exclusive use by an application or wishes to make a set of volumes available to a select group of users, it is appropriate to create a new pool to contain the volumes.

Often a central pool root/public is provided by the Media Administrator for use by the general user population. Providing volumes centrally saves time spent on submission and retrieval.

Allowing users to allocate volumes from another user's pool creates the situation where one person owns a physical volume while another person owns the data it contains. To maintain data security, DeepSpace does not give the volume owner any special privilege with respect to accessing the volume. Further, the volume owner may not retrieve the volume until it is scratched by the data owner. In general, a user may only retrieve a volume if it belongs to one of their pools. The only way to retrieve a volume from someone else's pool is to have an operator edit the volumeset into a pool owned by the user.

8.3 Options

Argument	Valid Value
keywords	One or more keyword=value assignments. There are no specified defaults associated with these keywords. See Keywords (see page 32)for a list of all valid keywords.
pool	Volume Pool Name. To reference a pool that belongs to someone else, the pool name is prefixed with the user ID of the pool's owner (i.e. root/public). The pool name may be up to twelve characters long.

8.4 Keywords

Argument	Valid Value
`padd`	= Automatic Volume Addition Flag. If set to yes, and a volume allocation request cannot be satisfied by an existing volume, the operator is prompted to provide a new volume. If the operator refuses, the allocation request fails. If the operator does provide a new volume, the volume is dynamically added to the catalog. (This keyword is available to operators only). For example: `padd=yes`
`pclean`	= Pool Cleaning Flag. If yes, whenever a pool volume is scratched, DeepSpace checks to see if the volume's cleaning count exceeds the cleaning threshold configured for the volume's Media Type. If the cleaning count is too high, the volume is scheduled for cleaning before being scratched. For example: `pclean=yes`
`gacc`	= Group Access List. Same as the "uacc" keyword only with group IDs. For example: `gacc=ANY,-guest`
`phold`	= Volume Hold Period. Set to an integer representing the number of days to hold a volume in limbo between when a user scratches it and when it is available for allocation again. This feature allows users to change their mind about scratching a volume. For example: `phold=10`

Argument	Valid Value
`uacc`	= User Access List. A comma-separated list of user IDs identifying who can allocate volumes from the pool. The special value ANY specifies any user. User IDs may be explicitly excluded by prefixing them with `-` (dash). For example: `uacc=ANY,-hacker`
`premove`	= Pool Removal Flag. If set to yes, whenever a pool volume is scratched, DeepSpace checks to see if the volume's usage count exceeds the usage threshold configured for the volume's Media Type. If the usage count is too high, the volume is scheduled for removal from the catalog. For example: `premove=yes`

8.5 See Also

- dsreport(1)(see page 46)
- dspedit(1)(see page 35)
- dspdelete(1)(see page 34)

9 dspdelete(1)

Name

`dspdelete` — delete a volume pool.

9.1 Synopsis

`dspdelete` *`pool`*

On this page:

9.2 Description

`dspdelete` deletes the specified pool. A pool cannot be deleted if it contains volumes. Only the pool owner is permitted to delete it.

9.3 Options

Argument	Valid Value
`pool`	Volume Pool Name. To reference a pool that belongs to someone else, the pool name is prefixed with the user ID of the pool's owner (for example, `root/public`).

9.4 See Also

dsreport(1)(see page 46)

dspedit(1)(see page 35)

dspcreate(1)(see page 31)

10 dspedit(1)

Name

`dspedit` — edit a volume pool.

10.1 Synopsis

`dspedit [`*`keywords`*`] `*`pool`*

On this page:

10.2 Description

`dspedit` modifies the attributes of the specified pool.

10.3 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. There are no specified defaults associated with these keywords. See Keywords (see page 35)for a list of all valid keywords.
`pool`	Volume Pool Name. To reference a pool that belongs to someone else, the pool name is prefixed with the userid of the pool's owner (i.e. root/public).

10.4 Keywords

Argument	Valid Value
`padd`	= Automatic Volume Addition Flag. If yes, and a volume allocation request cannot be satisfied by an exiting volume, the operator is prompted to provide a "new" volume. If the operator refuses, the allocation request fails. If the operator does provide a new volume, the volume is dynamically added to the catalog. (This keyword is available to operators only). For example: `padd=yes`
`pclean`	= Pool Cleaning Flag. If yes, whenever a pool volume is scratched, DeepSpace checks to see if the volume's cleaning count exceeds the cleaning threshold configured for the volume's Media Type. If the cleaning count is too high, the volume is scheduled for cleaning before being scratched. For example: `pclean=yes`
`agacc`	= A comma separated list of group IDs to add to the group access list.
`rgacc`	= A comma separated list of group IDs to remove from the group access list.
`phold`	= Volume Hold Period. Set to an integer representing the number of days to hold a volume in limbo between when a user scratches it and when it is available for allocation again. This feature allows users to change their mind about scratching a volume. For example: `phold=10`
`auacc`	= A comma-separated list of user IDs to add to the user access list.
`ruacc`	= A comma-separated list of user IDs to remove from the user access list
`premove`	= Pool Removal Flag. If set to yes, whenever a pool volume is scratched, DeepSpace checks to see if the volume's usage count exceeds the usage threshold configured for the volume's Media Type. If the usage count is too high, the volume is scheduled for removal from the catalog. For example: `premove=yes`

10.5 See Also

dsreport(1)(see page 46)
dspdelete(1)(see page 34)
dspcreate(1)(see page 31)

11 dspretrieve(1)

Name

`dspretrieve` — initiate retrieval of a scratch volume.

11.1 Synopsis

`dspretrieve` *`vid`* `[nvol=`*`n`*`]`

On this page:

11.2 Description

`dspretrieve` assigns the named volume a receipt number. To complete retrieval, present the receipt number to the operator in exchange for the volume.

If DeepSpace is not configured to require volume acceptance, or, *`vid`* has not yet been accepted, the corresponding volume record is deleted from the catalog, and retrieval is complete. A receipt number is not generated.

If acceptance is required and *`vid`* has been accepted, DeepSpace assigns the volume a receipt number.

Only the owner may retrieve a volume.

11.3 Options

Argument	Valid Value
`vid`	Volume ID. A unique identifier assigned to each volume when it is submitted.
`nvol`	=Number of tapes. Specifies the number of tapes to be selected. The default is 1.

11.4 See Also

dspsubmit(1)(see page 39)

12 dspsubmit(1)

12.1 Name

`dspsubmit` — submit a scratch volume to DeepSpace.

12.2 Synopsis

`dspsubmit` *`keywords`*

On this page:

12.3 Description

`dspsubmit` initiates submission of a scratch volume to DeepSpace. Scratch volumes are volumes that do not belong to a volumeset. Scratch volumes join volumesets either when allocated by the `dsvcreate(1)` command or when they are used to extend an existing volumeset when `dsvwrite(1)` drafts a new volume from a pool.

The `dspsubmit` keywords describe attributes of the volume such as length, media type, etc. If a keyword is not given, the corresponding attribute is given a default value from the Default Definition Record that matches the indicated media type.

Upon successful completion, `dspsubmit` displays the volume ID assigned to the volume. The volume ID is a unique name for the volume within the database. It should be written on an external label affixed to the volume.

After `dspsubmit` completes, the catalog contains a record for the volume. If DeepSpace is configured to require volume acceptance and/or volume identification, the volume is not available for use until it is handed over to an operator who completes the submission process.

12.4 Options

Argument	Valid Value
keywords	One or more keyword=value assignments. There are no specified defaults associated with the following keywords. See Keywords[2]for a list of all valid keywords.

12.5 Keywords

Argument	Valid Value
`app`	= Application prefix. Used with the volume and file listing reports. The `app` keyword limits selection to items that begin with the given string. For example: `app=89`
`capacity`	= Volume capacity in Mbytes. Specifies the volume capacity for cartridge media types.
`ecnt`	= Error count. Set error count as specified.
`finger`	= File fingerprint. Set fingerprint field as specified.
`flocation`	= Free volume location. When a volumeset is truncated or scratched, the newly unattached volumes will return to the location assigned to `flocation`. The value given must be defined as a DeepSpace storage site (the command `dsreport --sites` will produce a list of currently defined storage sites).
`location`	=Volume location. The site where the new volume is located. The value must be defined as a DeepSpace storage site (the command `dsreport --sites` produces a list of currently defined storage sites). This keyword sets the sloc, cloc, and floc fields for the volume unless those fields are also specified on the command line.
`nvol`	= Number of tapes. Specifies the number of tapes to submit to the library. The *`vid`*, *`rack`* and *`vsn`* values are incremented for each repetition. The default is 1.

2 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319225884#dspsubmit(1)-KeywordSection

Argument	Valid Value
`vsn`	= Volume Serial Number. The volume serial number (VSN) associated with the volume. The VSN is written in the VOL1 label for IBM and ANSI tape formats. Even unlabeled volumes (IBMU, TAR, CPIO and RAW) must be given a VSN. In general DeepSpace does not require each volume to have a unique VSN. The VSN may be up to six characters long. The special value `vms` means that this value is selected by the Vault Management System.
`type`	= Volume Media Type. Media type is an arbitrary name assigned during DeepSpace configuration to describe Volume formats supported by devices under DeepSpace control. For 9-track devices, this is typically the tape density (i.e. 1600, 6250, etc.). For cartridge devices, it is typically the cartridge standard (i.e. QIC11, QIC24, etc.). The given value must be a defined Media Type. The command `dsreport --types` produces a list of defined Media Types.
The following keywords may only be used by DeepSpace operators.	
`clean`	= Volume cleaning count. The number of times the volume has been mounted since it was last cleaned. This value is normally maintained by DeepSpace.
`clocation`	= Current Volume Location. The value given must be defined as a DeepSpace storage site. (The `dsreport --sites` command produces a list of currently defined storage sites.)
`ctype`	= Current Media Type. For initialized volumes, this field contains the Media Type DeepSpace has verified is the correct media type for the volume. This field is normally maintained by DeepSpace.
`rack`	=Rack Number. The rack number identifying the volume's storage slot. The `rack` keyword is provided here so operators can submit volumes in a single step. It can be up to twelve characters long. The special value `vms` means that this value is selected by the Vault Management System. If volume acceptance is required, the rack number is normally assigned when the operator accepts the volume.
`vid`	= Volume ID. A unique volume identifier. For sites that maintain unique Volume Serial Numbers, `vid` is normally identical to `vsn`. By default, DeepSpace, assigns each volume a unique volume ID by combining a four character word with a three digit number: for example, root-496. It can be up to twelve characters long. The special value `vms` means that this value is selected by the Vault Management System. The VID may not begin with a period (".") or contain a colon (":").

Argument	Valid Value
`maintenance`	= Scheduled Maintenance. Maintenance may be set to one or more of the folllowing: • move - indicates the volume is awaiting movement. • age - causes the volume to wait *N* days after it is scratched before becoming free. (*N* is defined by the pool the tape belongs to.) • erase - schedules the volume for erasure. • remove - • clean - schedules the volume for cleaning. Multiple items are selected by separating them with the `&` (ampersand) character. For example, `maint="erase&clean"` Normally the maintenance value is maintained by DeepSpace.
`remove`	= Volume Usage Count. The number of times the volume has been mounted since it entered the catalog. This value is normally maintained by DeepSpace.
`pool`	= Pool membership. Every volume must belong to a pool. By default, volumes belong to the user's private pool. Volumes may only be assigned to pools owned by the current effective user ID. The full specification of pool is `userid/pool`. If `userid` is omitted, the current effective user ID is assumed. Only operators may create pools for IDs other than their own.
`status`	= Volume submission status. Status may be assigned the following. • submitted awaiting acceptance. • accepted awaiting identification. • fully submitted. • retrieved awaiting return. Submission status is normally maintained by DeepSpace.

12.6 See Also

dsedit(1)(see page 12)
dspretrieve(1)(see page 37)

13 dsrcreate(1)

Name

`dsrcreate` — create a rotation schedule.

13.1 Synopsis

`dscreate` *`rotsched`*

On this page:

13.2 Description

`dsrcreate` creates a rotation schedule named *`rotsched`*. A rotation schedule is a list of one or more “location-type-count” triples.

Volumesets can be assigned rotation schedules. When a volumeset is created and assigned a rotation schedule, it moves to the first location in the list, remaining there for the time designated by the type-count. After that period has passed, the volumeset then moves to the second location. This process continues until the volumeset has proceeded through the entire list—at which point the volumeset expires.

`dsrcreate` conducts a dialog with the user repeating requests asking for the location-type-count specification for each step of the schedule. When the entire schedule has been entered, the user enters nothing at the next prompt to finish.

Durations can be specified two ways:

- R – indicates that the count is a fixed number of days the volumeset should stay at the location - example: "onsite R 30". After the "count" days have occurred, the volumeset moves to the next step of the schedule.
- G – indicates that the count represents the number of generations of the volumeset which should be at this step of the schedule. That is, only "count" generations may reside at the step. When a new generation arrives at the step, the oldest generation leaves the step for the next step in the schedule. When the oldest generation leaves the last step, it expires.

ⓘ You cannot use both R and G types in the same schedule.

13.3 Options

Argument	Valid Value
`rotsched`	A list of one or more “location-type-count” triples. Volumesets can be assigned rotation schedules....

13.4 Examples

```
dsrcreate sched1 Enter {G|R} :onsite G 1 Enter {G|R} :offsite G 2 Enter {G|R} :
```

In this example, the `sched1` schedule has two steps. The first step has one generation residing at the location `onsite`. The second step has two generations residing at the location `offsite`. The chronology is as follows:

- Generation 1 is born and resides at onsite until Generation 2 is born. At that point, Generation 1 moves to offsite and Generation 2 takes up residence at onsite.
- When Generation 3 is born, Generation 2 moves to offsite to reside with Generation 1 since this step can accommodate two generations.
- Finally, when Generation 4 is born, Generation 3 moves to offsite, and Generation 1 expires since there is only room for two generations at offsite - in this case Generation 2 and Generation 3.

The `dsvcreate(1)` command describes how to create multiple generations of the same volumeset.

13.5 See Also

dsrdelete(1)(see page 45)
dsvcreate(1)(see page 63)

14 dsrdelete(1)

14.1 Name

`dsrdelete` — delete a rotation schedule.

14.2 Synopsis

`dsrdelete` *`rotsched`*

On this page:

14.3 Description

`dsrdelete` deletes the named rotation schedule. Any volumesets under the schedule henceforth behave as if they have no rotation schedule.

14.4 Options

Argument	Valid Value
`rotsched`	A list of one or more “location-type-count” triples. Volumesets can be assigned rotation schedules....

14.5 See Also

dsrcreate(1)(see page 43)

15 dsreport(1)

15.1 Name

`dsreport` — DeepSpace report generator.

15.2 Synopsis

```
dsreport --report  {dinfo | finfo | flist | pinfo | plist | pvolumes | rinfo | rlist |
sites | types | vinfo | vlist | vflist | vsflist | vslist | vsvlist}
```

On this page:

15.3 Description

`dsreport` generates the requested DeepSpace report. DeepSpace user reports are a subset of the DeepSpace operator reports. The operator reports are listed in the `dsreport(8)` manpage.

15.4 Options

Argument	Valid Value
report	Some reports require keyword= arguments to identify the item being reported. See Keywords[3]for a list of all valid keywords.
bdev	
crash	

3 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection

Argument	Valid Value
`dinfo`	Displays the configuration for the drive specified with the `adn` keyword. The following keywords are accepted: • adn • full See Keywords[4]for an explanation of each.
`finfo`	Displays the full catalog entry for the tape file identified by the fid keyword. The following keywords are accepted: • fid • fseq • fsect • full • volset See Keywords[5]for an explanation of each.
`flist`	Lists all the files owned by the user. The following keywords are accepted: • app • full See Keywords[6]for an explanation of each.
`rinfo`	Lists rotation schedule information. The following keywords are accepted: • full • rotation See Keywords[7]for an explanation of each.
`rlist`	Lists rotation schedules.The following keyword is accepted: full. See Keywords[8]for a description.

4 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
5 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
6 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
7 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
8 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection

Argument	Valid Value
`maint`	XXXX. The following keywords are accepted: • action • location • full See Keywords[9]for an explanation of each.
`sinfo`	Displays the information of the specified library site. The following keyword is accepted: • sid • full full. See Keywords[10]for a description.
`slist`	Lists all library sites. The following keyword is accepted: • user • full See Keywords[11]for a description.
`sites`	Lists all library sites. The following keyword is accepted: • full See Keywords[12]for a description.
`pinfo`	Displays the catalog entry for the pool named with the pool keyword. The following keywords are accepted: • pool • full See Keywords[13]for an explanation of each.

9 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
10 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
11 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
12 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
13 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection

Argument	Valid Value
`plist`	Lists all pools owned by the user. The following keyword is accepted: • full See Keywords[14]for a description.
`pvolumes`	Lists a summary of volumes currently assigned to the specified pool. The following keywords are accepted: • full • pool See Keywords[15]for an explanation of each.
`tapecap`	Lists all defined media types.The following keyword is accepted: • entry See Keywords[16]for a description.
`types`	Lists all defined media types.The following keyword is accepted: • full See Keywords[17]for a description.
`vinfo`	Displays the catalog entry for the volume named by the volset or vid keywords. The following keywords are accepted: • full • vid • volset See Keywords[18]for an explanation of each.
`vlist`	Lists a summary of all selected volumes. The following keyword is accepted: full. See Keywords[19]for a description.

14 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
15 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
16 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
17 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
18 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
19 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection

Argument	Valid Value
`vflist`	Lists all files on the volume named by the vid keyword. The following keywords are accepted: • full • vid See Keywords[20]for an explanation of each.
`vsflist`	Lists all files on the volumeset named by the volset keyword. The following keywords are accepted: • full • volset See Keywords[21]for an explanation of each.
`vslist`	Lists all volumesets belonging to the user. The following keywords are accepted: • app • full • pexpire See Keywords[22]for an explanation of each.
`vsvlist`	Reports the member volumes of the volumeset named by the volset keyword. The following keywords are accepted: • full • volset See Keywords[23]for an explanation of each.

15.5 Keywords

Argument	Valid Value
`--action`	

20 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
21 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
22 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection
23 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=319160385#dsreport(1)-KeywordSection

Argument	Valid Value
`--adn`	=the drive's ADN. Used with the `dinfo` report. For example: `--adn=drive1`
`--app`	= Application prefix. Used with the volume and file listing reports, the app keyword limits selection to items that begin with the given string. For example: `--app=89`
`--crloc`	
`--days`	
`--entry`	= name specified in the tapecap file (e.g. LTO)
`--fid`	= File ID. Identifies the file for the `finfo` report.
`--fsect`	=File section number. Identifies the file by the file section number for the `finfo` report.
`--fseq`	= File sequence number. Identifies the file by order in the volumeset for the finfo report.
`--full`	= Specifies full report format or parsable format (with fields delimited by colons). The default is `--full=yes`, which specifies full format; The value `--full=no` specifies parsable format.
`--host`	= Specifies host to retrieve information from.
`--location`	= Specifies name of location
`--pool`	= Pool Name. Used with the `pinfo`, `vinventory`, and `pvolumes` reports. For example: `--pool=private`
`--rack`	= Specifies rack id
`--rotation`	= Rotation schedule. Used with the `rinfo` report.
`--sid`	= site id

Argument	Valid Value
`--user`	= operator user name as specified in /etc/passwd file
`--vid`	= Volume ID. Identifies the volume for the `vinfo` and `vflist` reports.
`--volset`	= Volumeset name. Identifies the volumeset for the `vinfo`, `finfo`, `vsflist`, and `vsvlist` reports. For example: `--volset=volset1`
--pexpire	= Specifies whether to print expiration info or not. The default is --pexpire=no.

15.6 See Also

- How to Request a Volumeset Volume List User Report[24]
- How to Request a Volumeset List User Report[25]
- How to Request a Volumeset File List User Report[26]
- How to Request a Volumeset Attributes User Report[27]
- How to Request a Volume List User Report[28]

24 https://deepspace.atlassian.net/wiki/spaces/DeepSpaceDoc/pages/287604843/How+to+Request+a+Volumeset+Volume+List+User+Report
25 https://deepspace.atlassian.net/wiki/spaces/DeepSpaceDoc/pages/284688396/How+to+Request+a+Volumeset+List+User+Report
26 https://deepspace.atlassian.net/wiki/spaces/DeepSpaceDoc/pages/287703143/How+to+Request+a+Volumeset+File+List+User+Report
27 https://deepspace.atlassian.net/wiki/spaces/DeepSpaceDoc/pages/284524566/How+to+Request+a+Volumeset+Attributes+User+Report
28 https://deepspace.atlassian.net/wiki/spaces/DeepSpaceDoc/pages/284753931/How+to+Request+a+Volume+List+User+Report

16 dsreserve(1)

16.1 Name

`dsreserve` — reserve one or more devices.

16.2 Synopsis

`dsreserve [`*`keywords`*`] [`*`psd`*`[,`*`psd...`*`]]`

On this page:

16.3 Description

`dsreserve` asks the resource manager to reserve one or more devices. When `dsreserve` completes, the requested devices are available for use. If the request cannot be satisfied, `dsreserve` prints an appropriate error message.

To avoid deadlocks, a user who currently has a reserved device may not reserve a second device. Users who require simultaneous access to multiple devices, must request all devices with a single `dsreserve` command.

The type of device to reserve may be specified a variety of way"

- `adn=`*adn_list* allows specification of the exact names of the device(s) to reserve.
- `type=`*type_list* is a less specific (and therefor easier to satisfy) request for any device capable of supporting the indicated Media Type(s).
- `volset=`*volset_list* requests device(s) compatible with the given volumeset(s). Only one of `adn`, type or `volset` may be specified.

Unless a device is reserved by adn, the actual device assigned to a reservation may change. For example, the second volume of a volumeset may be mounted on a different device while the first volume is rewinding.

16.4 Options

Argument	Valid Value
keywords	One or more keyword=value assignments. See Keywords (see page 54)for a list of all valid keywords.
[*psd*[,*psd...*]]	Pseudo Name List. Comma separated list of Pseudo Device Names. Reservation requests for more than one device must provide each device a Pseudo Device Name. Pseudo Device Names are used to distinguish the different devices after they are reserved. If no Pseudo Device Name is given on a single device reservation request, the name "default" is assumed. Each name can be up to twelve characters long.

16.5 Keywords

Argument	Valid Value
adn	= ADN List. Comma separated list of ADNs to reserve. There is no specified default.
type	= Media Type List. Comma separated list of Media Types to reserve. There is no specified default.
volset	= Volumeset List. Comma separated list of volumeset names. The resource allocator checks the Media Type associated of each volumeset and reserves a compatible device. There is no specified default.
mach	= Machine. Use a drive on the designated machine. If the machine is unspecified and the user's machine has an appropriate drive, then by default the request is issued for service by the user's machine. If there is no drive available on the user's machine, then the request will fail and an error message is returned. A value of any will issue a request for any appropriate drive in the network.
queue	= Reservation Queue Flag. If set to yes and the request cannot be satisfied immediately, the device reservation request is queued. The default is to return an error.
priority	= Reservation Queue Priority. An integer value from 0 to 9 used to order reservation requests in the queue. Lower priority requests are serviced first. The default priority is 5.

Argument	Valid Value
`idle`	= Idle Device Flag. By default, if a reserved device is idle for a configurable length of time, DeepSpace cancels the reservation and makes the device available for use by others. Setting `idle` to yes prevents DeepSpace from canceling a reservation because of excessive idle time. The default is no. ⓘ The time-out period between `dsreserve` and `dsvmount` is short; therefore setting `idle=yes` is recommended.

16.6 See Also

dsvmount(1)(see page 79)
dsstatus(1)(see page 57)
dsfree(1)(see page 25)

17 dss(1)

17.1 Name

`dss` — DeepSpace server summary.

17.2 Synopsis

```
dss [-q  -c] [ -d ]
```

On this page:

17.3 Description

`dss` prints a summary of the environment in which the DeepSpace servers are running. It also pings each server and reports whether the server is responding.

-q - specifies display request queue only.

-c - when specified with -q returns only the number of requests in the queue

-d - specifies display only the device status's

No options returns both the request queue list and device status list.

17.4 See Also

- ds(8)[29]
- DS(8), DSexit(8), DStest(8)[30]
- DeepSpace Catalog Data Services Operations Guide[31]

29 https://deepspace.atlassian.net/wiki/spaces/DeepSpaceDoc/pages/314441732
30 https://deepspace.atlassian.net/wiki/spaces/DeepSpaceDoc/pages/348782620
31 https://deepspace.atlassian.net/wiki/spaces/DeepSpaceDoc/pages/247922689/DeepSpace+Catalog+Data+Services+Operations+Guide

18 dsstatus(1)

18.1 Name

`dsstatus` — display device reservation status.

18.2 Synopsis

`dsstatus`

On this page:

18.3 Description

If a user has a queued resource request, `dsstatus` displays the resource request queue. If a user has reserved devices, `dsstatus` displays information about the devices under the user's control.
The report has two sections: Request Queue and Device Status.

Request Queue Section		Device Status Section	
UID	The user's name.	Pseudo	The pseudonym assigned to the drive - see the `dsvaccess(1)` command description.
Prio	The request's priority - a digit 0 through 9. Lower digits have higher priority.	Type	The media type involved in the request.

Request Queue Section		Device Status Section	
Key	The request's key name. Usually this is the same as the user's name. It can be different, see the `dsvaccess(1)` command description.	Stat	The status indicator: • vrfy - the mounted volume is being electronically identified. • on - a volume is mounted but not under user control. • off - the drive is off-line; there is not a mounted volume. • dchk - the drive is undergoing a density check. • user - the mounted volume is under user control. • rew - the mounted volume is being rewound prior to being unmounted. • prem - a volume has been premounted - it is unknown the volume's disposition for write protection. • wprem - a volume has been premounted for reading or writing. • rprem - a volume has been premounted for reading only. • octl - the drive is about to enter the off state.
Machine	The request originated on the named computer.	VID	The volume ID currently mounted on the drive.
Format	The media type involved in the request.	Volumeset	The name of the volumeset in use.
Adn	If a specific drive was requested, it is listed here.		
Psd	The pseudonym assigned to the drive - see the `dsvaccess(1)` command description.		

18.4 See Also

dsvaccess(1)(see page 61)
dsvrelease(1)(see page 87)

19 dsunq(1)

19.1 Name

`dsunq` — unqueue DeepSpace requests.

19.2 Synopsis

`dsunq [key=keyname] [user=uname]`

On this page:

19.3 Description

`dsunq` cancels the outstanding user request. User's can review their outstanding requests with the `dsstatus(1)` command.

19.4 Options

Argument	Valid Value
keywords	One or more keyword=value assignments. See Keywords[32]for a list of all valid keywords.

19.5 Keywords

Argument	Valid Value
key	= keyname. The name under which the request was made. dsunq assumes this is the user's name unless this is specified.

32 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=323846188#dsunq(1)-KeywordSection

Argument	Valid Value
`user`	= uname. This keyword directs the program to work with the request associated with the specified user name. This keyword can only be used by operators.The default is the current effective user ID.

19.6 See Also

dsstatus(1)(see page 57)

20 dsvaccess(1)

20.1 Name

`dsvaccess` — initiate access to a volumeset.

20.2 Synopsis

`dsvaccess [`*`keywords`*`] `*`volset`*` [`*`psd`*`]`

On this page:

20.3 Description

`dsvaccess` initiates access to a volumeset on device *psd*. If no device is reserved as *psd*, `dsvaccess` reserves one.

Accessing a volumeset does not immediately cause a mount; it just confirms the following:

- The volumeset is not locked (that is, in use elsewhere).
- The user has the necessary permission.
- A suitable device is available.

A mount request is generated the first time the volumeset is read or written.

When using this command, you must first use the `dsreserve` command to reserve one or more devices. Unless the device is reserved by ADN, the actual device assigned to the reservation may change.

20.4 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. See Keywords (see page 62)for a list of all valid keywords.

Argument	Valid Value
volset	Volumeset Name. Either the name given to the volumeset when it was submitted or '.' followed by the volume ID of the first volume in the volumeset.
psd	Pseudo Device Name. An arbitrary name assigned during device reservation; used to distinguish between devices when multiple devices are reserved.

20.5 Keywords

Argument	Valid Value
`machine`	= Machine. Use a drive on the designated machine. If the machine is unspecified and the user's machine has an appropriate drive, then by default the request is issued for service by the user's machine. If there is no drive available on the user's machine, then the request will fail and an error message is returned. A value of `any` will issue a request for any appropriate drive in the network.
`priority`	= Reservation Queue Priority. An integer value from 0 to 9 used to order reservation requests in the queue. Lower priority requests are serviced first. The default priority is 5.
`queue`	= Reservation Queue Flag. If set to yes and the request cannot be satisfied immediately, the device reservation request is queued. The default is to return an error.
`passwd`	= Volume Access Password. An optional password which, if specified, must be given before the volumeset can be read or written. There is no specified default.
`write`	= Write Access. If yes, a volumeset can be accessed in write mode. The default is no.

20.6 See Also

dsvcreate(1)(see page 63)
dsvrelease(1)(see page 87)

21 dsvcreate(1)

21.1 Name

`dsvcreate` — create a volumeset.

21.2 Synopsis

`dsvcreate [`*`keywords`*`] [`*`volset`*`]`

On this page:

21.3 Description

`dsvcreate` allocates a scratch volume from the pool indicated by the pool keyword. The volume is converted into a volumeset with attributes specified by the command keywords. If a keyword is not given, the corresponding attribute is given a default value. See `dsvsubmit(1)` for a description of how default values are selected. Upon successful completion, `dsvcreate` displays the volume ID of the allocated volume.

21.4 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. There are no specified defaults associated with these keywords except where noted. See Keywords (see page 64)for a list of all valid keywords.

Argument	Valid Value
volset	Volumeset Name. A meaningful name assigned by the volumeset owner. The general form of *volset* is `[userid/]vname[:Ggno][:Vvno]`.In this syntax: • *userid* is a valid user ID. If userid is omitted, the effective user ID is assumed. Only an operator may submit to a *userid* other than the current effective user ID. • *vname* is an arbitrary string up to 64 characters long. • *Ggno* specifies a volumeset generation number. *gno* can be an integer or a signed integer. If *gno* is an integer, it references the given generation number. If *gno* is a signed integer, it references the highest existing generation number offset by *gno*. For example, if the highest generation number for volumeset test is 10, :G-1 refers to generation 9 and :G+1 refers to generation 11. If the generation specification is omitted, the highest existing generation is referenced. If there are no existing generations, generation 0 is selected. The highest allowable generation number is 9999. • *Vvno* specifies a volumeset version number. Generation and version numbers are used as subscripts for volumesets with the same *vname*. Version numbers behave like generation numbers. The highest allowable version number is 99. If *volset* is omitted, an unnamed volumeset is created. Unnamed volumesets are referenced by a period (`.`) followed by the volume ID of the first volume in the volumeset. For example, If the first volume has volume ID 123456 the volumeset is referenced as `.123456`.

21.5 Keywords

Argument	Valid Value
`app`	= Application prefix. This prefix is used with file reports to limit selections to those with the specified `app`= value. `app` may be up to 12 characters long.
`capacity`	= Volume capacity in Mbytes. Specifies the volume capacity for legacy cartridge media types, and even then is immaterial for media that is only read to bring archived data into the data catalog. Where this attribute IS used is for archive systems that have a requirement for keeping individual objects below a minimum size so that the back-end does not suffer from a "lumpy" data distribution, i.e., CEPH clusters accessed through the librados API have a limited maximum file size before requiring the remainder spill over into another object.

Argument	Valid Value
`conv`	= Record conversion specification. conv controls conversion of records to/from tape. conv may be set to one of the following values: • text for text record: • etext for EBCDIC text records. • data for fixed length ASCII or binary data records. • edata for fixed length EBCDIC data records.
`dispose`	= Volume disposition. dispose controls when volumes leave the volumeset and what happens when they do. dispose may be set to erase and/or retain. If erase is specified, volumes that leave the volumeset (when it is truncated or scratched) are erased before they can be reallocated. If retain is specified, the volumeset is never truncated. (Truncation occurs when the first file on a volumeset is overwritten and the new file is not large enough to span all the volumes currently in the volumeset.) To select both, use the following setting, including the quotation marks: `dispose="erase&retain"`
`ecnt`	= Error count. Set error count as specified.
`finger`	= File fingerprint. Set fingerprint field as specified.
`flocation`	= Free volume location. When a volumeset is truncated or scratched, the newly unattached volumes will return to the location assigned to flocation. The value given must be defined as a DeepSpace storage site (the `dsreport --sites` command will produce a list of currently defined storage sites). The value can be up to twelve characters long.
`format`	= Label Format. format may be set to one of the following values: • DSCM • ANSI • IBM • IBMU • TAR • CPIO • RAW

Argument	Valid Value

`ftemplate`	= File Name template. The filename template is used to dynamically construct names for files written to the volumeset. The name constructed by the template is limited to seventeen (17) characters. The template consists of constant text and substitution patterns. Substitution patterns have the general form:`@sublen@`. In this syntax,
	
	• @ delimits the beginning and end of the substitution specification.
	• *`sub`* is a character indicating what to substitute.
	• *`len`* gives the length in characters of the substituted value. In principle it can be any number between 1 and 17. However, for some values of sub only a particular value of *len* is reasonable. The following values are recognized.
	  • @Y4@ – numeric year.
	  • @C2@ – numeric month (Jan = 1).
	  • @E3@ – month name.
	  • @D2@ – day of month.
	  • @J3@ – Julian day.
	  • @W3@ – day of week (Sun = 1).
	  • @H2@ – hour (24 hour clock).
	  • @M2@ – minute.
	  • @S2@ – second.
	  • @F9@ – value assigned fid keyword on the `dsvwrite` command.
	  • @U9@ – base name of file assigned to the if keyword on the `dsvwrite` command.
	  • @G4@ – file generation.
	  • @V2@ – file version.
	
	Numeric values are truncated on the left. Character values are truncated on the right. If a numeric value is shorter then len it is padded on the left with zeros. For example, on January 25 2018: ftemplate=GV, results in the following file name:
	
	`25Jan2018G0000V01`

`ftrack`	= File Tracking Flag. If set to yes, this causes the catalog to maintain a record for every file written to the volumeset. If set to no, this disables file cataloging. Maintaining file catalogs makes it possible to generate a volumeset table of contents without having to mount a tape. It also improves the efficiency of accessing files on multi-volume volumesets.
`location`	=Volume location. The site where the volumes for the new volumeset are located. The value must be defined as a DeepSpace storage site (the `dsreport --sites` command lists the currently defined storage sites).
`offset`	= Record offset. The number of bytes reserved at the beginning of each block for additional information. Offset only affects the ANSI tape format.

Argument	Valid Value
`passwd`	= Volume Access Password. An optional password which, if specified, must be given before the volumeset can be read or written. It can be up to fourteen characters long.
`pool`	= Pool Source. The name of the pool from which to draft tapes into the volumeset. The default is the user's private pool. For example: `username/private`

`rformat`	= Record Format. rformat has the general form: fmt:blen:rlen. In this syntax,
	
	• fmt is the record format:
	  • f - fixed length records.
	  • fb - fixed length, blocked records.
	  • v - variable length records.
	  • vb - variable length, blocked records.
	  • vs - variable length, spanned records.
	  • vbs - variable length, blocked, spanned records.
	  • u - unformatted data.
	• blen is the block length in bytes.
	• rlen is the record length in bytes.
	
	For example:
	
	`rformat=fb:800:80`
	
	The default is u.

`scratch`	= Scratch status. Sets volume scratch status. (If volume is the first member of the volumeset, scratches the entire volumeset.)
`slocation`	= Scheduled Volume Location. If a volume's current location (clocation) does not equal its scheduled location, the volume will show up on the Volume maintenance report as wanting to move from clocation to slocation. The value given must be defined as an DeepSpace storage site. (The `dsreport --sites` command will produce a list of currently defined storage sites.) It can be up to twelve characters long.
`type`	= Volume Media Type. Media type is an arbitrary name assigned during DeepSpace configuration to describe volume formats supported by archive devices under DeepSpace control. The given value must be a defined media type. The `dsreport --types` command produces a list of currently defined media types. It can be up to eight characters long. The default is configurable (the first site in the `dsconfig` site list).

Argument	Valid Value
`vaccess`	= Volume Access Byte. The character assigned to vaccess is written as byte 11 (Volume Accessibility) in the VOL1 label. DeepSpace attaches no particular significance to the Volume Access byte. Control is provided for export of volumes to sites that require certain values. This applies to IBM and ANSI formats only. The default is 0x00.
`vcomment`	= Volumeset comment. A comment about the volumeset. If the comment includes spaces, it must be enclosed in quotes. It can be up to 40 characters long. For example: `vcomment="my favorite volumeset"`
`vexpire`	= Volumeset Expiration Date. It can be one of the following values: • I - infinite (never expires) • S - scratch (immediately expired) • RN - expires N days after creation • AN - expires if not accessed in N days. • L - expires when all files on the volumeset have expired. • Orotsched - follows the rotation schedule rotsched (see `dsrcreate(1)`). • Xccyymmdd or Xmm/dd/yy - expires on given date. • GN - expire when there are N newer generations. For example: `vexpire=R30` The default is S. ⓘ Upon expiration, the volumeset is not disbanded until it is scratched. The `dsvscratch` program disbands volumesets.
`vmode`	= Volumeset Permission Mask. Three octal digits controlling owner, group and others (respectively) permission to the volumeset (similar to the UNIX file mode). Bit 4 controls read access. Bit 2 controls write access. Bit 1 permits viewing of the volume's catalog entry. For example: `vmode=744` The default is 700.
	The following keywords may only be used by DeepSpace operators:
`group`	= Volumeset Group. The group (from `/etc/group`) to which the volumeset belongs. It can be up to twelve characters long. The default is the current effective group ID.

Argument	Valid Value
`initialize`	= Volumeset Initialization. If set to yes, it indicates the volumeset requires initialization. The first time each volume is mounted, DeepSpace will initialize it. If set to no, it indicates the volumeset contains existing data. The first time each volume is mounted DeepSpace checks to make sure the Volume Serial Number on the volume label matches the Volume Serial Number in the catalog. The default is yes.
`maintenance`	=Scheduled Maintenance. Maintenance may be set to one or more of the following values: • move - indicates the volume is awaiting movement. • age - causes the volume to wait N days after it is scratched before becoming free (N is defined by the pool the volume belongs to). • erase - schedules the volume for erasure. • remove • clean - schedules the volume for cleaning. Multiple items are selected by separating them with `&` (ampersand). For example: `maint="erase&clean"` The default is no maintenance. Normally the maintenance value is maintained by DeepSpace.
`user`	= Volumeset Owner. The user ID (from `/etc/passwd`) to which the volumeset belongs. It can be up to twelve characters long. The default is the current effective user ID.

21.6 See Also

dsvsubmit(1)(see page 92)
dsvaccess(1)(see page 61)
dsvscratch(1)(see page 90)

22 dsvdisplay(1)

22.1 Name

`dsvdisplay` — display information about an accessed volumeset.

22.2 Synopsis

`dsvdisplay rep[ort]={info | volumes | toc | scan | labels {fseq=`*`fseq`*` | fid=`*`fid`*`}} [`*`psd`*`]`

On this page:

22.3 Description

`dsvdisplay` produces a variety of reports about the volumeset currently accessed via `dsvaccess(1)`.

> Do not interrupt. If the process is killed, you will need to use `dsvrelease` on the referenced volumeset.

22.4 Options

Argument	Valid Value
psd	Pseudo Device Name. An arbitrary name assigned during device reservation; used to distinguish between devices when multiple devices are reserved.

Argument	Valid Value
`rep[ort]`	=Report Name. Selects which report to generate. It can be one of the following values: • info - displays volumeset default values. • volumes - lists the volumes that comprise the volumeset. • toc - lists a table of contents for files on the volumeset. Only files that have been previously read or written are included. • scan - provides a table of contents like toc but scan actually mounts and scans each volume until the end of the volumeset is reached. • labels - displays label information for a particular volumeset file. Either fseq or fid must be given to identify which file to display.
`volumes`	Lists the volumes that comprise the volumeset.
`fseq`	=File Sequence Number. Selects a file by its relative position on the volumeset. For example, fseq=1 selects the first file, fseq=2 selects the second, etc.
`fid`	=File ID. Specifies a tape file name. The assigned value has the general form: *fname*[:G*gno*][:V*vno*]. I this syntax, • *fname* is the file name (or File ID), an arbitrary string up to 17 characters long. • G*gno* gives a file generation number. The maximum gno is 9999. • V*vno* gives a file version number. Generation and version numbers are used as subscripts for files with the same *fname*. The maximum vno is 99. Both *gno* and *vno* are non-negative integer values.

22.5 See Also

dsreport(1)(see page 46)

23 dsvedit(1)

23.1 Name

`dsvedit` — modify the catalog entries for volumes in a volumeset.

23.2 Synopsis

`dsvedit [`*`keywords`*`] `*`volset`*

On this page:

23.3 Description

`dsvedit` modifies the catalog entry associated with each volume in the specified volumeset to reflect the given keyword assignments. Only the *volset* owner may edit the volumeset.

23.4 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. The defaults associated with the following keywords are the current volume settings. See Keywords (see page 72)for a list of all valid keywords.
`volset`	Volumeset Name. Either the name given to the volumeset when it was submitted or a period (`.`) followed by the volume ID of the first volume in the volumeset.

23.5 Keywords

Argument	Valid Value
`app`	= Application prefix. This prefix is used with file reports to limit selections to those with the specified app= value. `app` may be up to 12 characters long.
`capacity`	= Volume capacity in Mbytes. Specifies the volume capacity for cartridge media types.
`convert`	= Record conversion specification. convert controls conversion of records to/from tape. convert may be set to one of the following values: • text for text record: • etext for EBCDIC text records. • data for fixed length ASCII or binary data records. • edata for fixed length EBCDIC data records.
`dispose`	= Volume disposition. dispose controls when volumes leave the volumeset and what happens when they do. dispose may be set to erase and/or retain. If erase is specified, volumes that leave the volumeset (when it is truncated or scratched) are erased before they can be reallocated. If retain is specified, the volumeset is never truncated. (Truncation occurs when the first file on a volumeset is overwritten and the new file is not large enough to span all the volumes currently in the volumeset.) To select both, use the following setting, including the quotation marks: `dispose="erase&retain"`
`finger`	= File fingerprint. Set fingerprint field as specified.
`flocation`	= Free volume location. When a volumeset is truncated or scratched, the newly unattached volumes will return to the location assigned to flocation. The value given must be defined as a DeepSpace storage site. (The `dsreport --sites` command will produce a list of currently defined storage sites.) The value can be up to twelve characters long.
`format`	= Label Format. format may be set to one of the following values: • ANSI • IBM • IBMU • TAR • CPIO • RAW

Argument	Valid Value

`ftemplate`	= File Name template. The filename template is used to dynamically construct names for files written to the volumeset. The name constructed by the template is limited to seventeen (17) characters. The template consists of constant text and substitution patterns. Substitution patterns have the general form:`@sublen@`. In this syntax,
	
	• @ delimits the beginning and end of the substitution specification.
	• *`sub`* is a character indicating what to substitute.
	• *`len`* gives the length in characters of the substituted value. In principle it can be any number between 1 and 17. However, for some values of *sub* only a particular value of *len* is reasonable. The following values are recognized.
	  • @Y4@ – numeric year.
	  • @C2@ – numeric month (Jan = 1).
	  • @E3@ – month name.
	  • @D2@ – day of month.
	  • @J3@ – Julian day.
	  • @W3@ – day of week (Sun = 1).
	  • @H2@ – hour (24 hour clock).
	  • @M2@ – minute.
	  • @S2@ – second.
	  • @F9@ – value assigned fid keyword on the `dsvwrite` command .
	  • @U9@ – base name of file assigned to the if keyword on the `dsvwrite` command.
	  • @G4@ – file generation.
	  • @V2@ – file version.
	
	Numeric values are truncated on the left. Character values are truncated on the right. If a numeric value is shorter then len it is padded on the left with zeros. For example, on January 25 1987: ftemplate=GV, results in the following file name:
	
	`25Jan89G0000V01`

`ftrack`	= File Tracking Flag. If set to yes, this causes the catalog to maintain a record for every file written to the volumeset. If set to no, this disables file cataloging. Maintaining file catalogs makes it possible to generate a volumeset table of contents without having to mount a tape. It also improves the efficiency of accessing files on multi-volume volumesets.
`offset`	= Record offset. The number of bytes reserved at the beginning of each block for additional information. Offset only affects the ANSI tape format.

Argument	Valid Value
`passwd`	= Volume Access Password. An optional password which, if specified, must be given before the volumeset can be read or written. It can be up to fourteen characters long. For example: `password=undertow` If passwd is specified with no value, the existing password is deleted.
`pool`	= Pool Source. Volumesets acquire additional tapes from their source pool. Use this keyword to move volumesets between pools. The full specification of pool is *userid*/*pool*. If *userid* is omitted, the current effective user ID is assumed. Only operators may create pools for IDs other than their own.

`rformat`	= Record Format. rformat has the general form: *`fmt:blen:rlen`*. In this syntax,
	
	• *fmt* is the record format:
	  • f - fixed length records.
	  • fb - fixed length, blocked records.
	  • v - variable length records.
	  • vb - variable length, blocked records.
	  • vs - variable length, spanned records.
	  • vbs - variable length, blocked, spanned records.
	  • u - unformatted data.
	• *blen* is the block length in bytes.
	• *rlen* is the record length in bytes.
	
	For example:
	
	`rformat=fb:800:80`

`scratch`	= Scratch status. Sets volume scratch status. Do not scratch individual volumes that are part of a volumeset. Use `dsvscratch` to scratch a volumeset.
`slocation`	= Scheduled Volume Location. If a volume's current location (clocation) does not equal its scheduled location, the volume will show up on the Volume maintenance report as wanting to move from clocation to slocation. The value given must be defined as an DeepSpace storage site. (The `dsreport --sites` command will produce a list of currently defined storage sites.) It can be up to twelve characters long.

Argument	Valid Value
`type`	= Volume Media Type. Media type is an arbitrary name assigned during DeepSpace configuration to describe volume formats supported by devices under DeepSpace control. For 9-track devices, this is typically the tape density (i.e. 1600, 6250, etc.). For cartridge devices, it is typically the cartridge standard (i.e. QIC11, QIC24, etc.). The given value must be a defined media type. The `dsreport --types` command produces a list of currently defined media types. It can be up to eight characters long.
`vaccess`	= Volume Access Byte. The character assigned to vaccess is written as byte 11 (Volume Accessibility) in the VOL1 label. DeepSpace attaches no particular significance to the Volume Access byte. Control is provided for export of volumes to sites that require certain values.
`valloc`	= yes | no. Specify whether a volume is allocated (`valloc=yes`) or not (`valloc=no`). The default is yes.
`vcomment`	= Volumeset comment. A comment about the volumeset. If the comment includes spaces, it must be enclosed in quotes. It may be up to forty characters long. For example: `vcomment="my favorite volumeset"`
`vexpire`	= Volumeset Expiration Date. It can be one of the following values: • I - infinite (never expires) • S - scratch (immediately expired) • RN - expires N days after creation • AN - expires if not accessed in N days. • L - expires when all files on the volumeset have expired. • Orotsched - follows the rotation schedule rotsched (see `dsrcreate(1)`). • Xccyymmdd or Xmm/dd/yy - expires on given date. • GN - expire when there are N newer generations. For example: `vexpire=R30`
`vmode`	= Volumeset Permission Mask. Three octal digits controlling owner, group and others (respectively) permission to the volumeset (similar to the UNIX file mode). Bit 4 controls read access. Bit 2 controls write access. Bit 1 permits viewing of the volume's catalog entry. For example, the following allows the volumeset owner all permissions, and the group and others read permission only: `vmode=744`
The following keywords may only be used by DeepSpace operators.	

Argument	Valid Value
`clean`	= Volume cleaning count. The number of times the volume has been mounted since it was last cleaned. This value is normally maintained by DeepSpace.
`clocation`	= Current Volume Location. The value given must be defined as a DeepSpace storage site (the `dsreport --sites` command produces a list of currently defined storage sites). It can be up to twelve characters long.
`ctype`	= Current Media Type. For initialized volumes, this field contains the Media Type DeepSpace has verified is the correct media type for the volume. This field is normally maintained by DeepSpace.
`group`	= Volumeset Group. The group (from `/etc/group`) to which the volumeset belongs. By default, group is the current effective group ID. It can be up to twelve characters long.
`initialize`	= Volumeset Initialization. If set to yes, it indicates the volumeset requires initialization. The first time each volume is mounted, DeepSpace will initialize it. If set to no, it indicates the volumeset contains existing data. The first time each volume is mounted DeepSpace checks to make sure the Volume Serial Number on the volume label matches the Volume Serial Number in the catalog.
`maintenance`	=Scheduled Maintenance. Maintenance may be set to one or more of the following values: • move - indicates the volume is awaiting movement. • age - causes the volume to wait N days after it is scratched before becoming free (N is defined by the pool the tape belongs to). • erase - schedules the volume for erasure. • remove • clean - schedules the volume for cleaning. Multiple items are selected by separating them with `&` (ampersand). For example: `maint="erase&clean"` The default is no maintenance. Normally the maintenance value is maintained by DeepSpace.
`remove`	= Volume Usage Count. The number of times the volume has been mounted since it entered the catalog. This value is normally maintained by DeepSpace.

Argument	Valid Value
`status`	= Volume submission status. Status may be assigned the following integer values. • 1 - submitted awaiting acceptance. • 2 - accepted awaiting identification. • 3 - fully submitted. • 4 - retrieved awaiting return. Submission status is normally maintained by DeepSpace.
`user`	= Volumeset Owner. The user ID (from `/etc/passwd`) to which the volumeset belongs. It can be up to twelve characters long. The default is the current effective user ID.

23.6 See Also

dsvsubmit(1)(see page 92)
dsvaccess(1)(see page 61)
dsvretrieve(1)(see page 88)

24 dsvmount(1)

24.1 Name

`dsvmount` — mount a volume.

24.2 Synopsis

`dsvmount [`*`keywords`*`] `*`volset`*` [`*`psd`*`]`

On this page:

24.3 Description

`dsvmount` asks the operator to mount volume *volset* on device *psd*. If no device is reserved as *psd*, `dsvmount` reserves one.

When a volume is mounted with `dsvmount`, interaction with the volume takes place directly through the `/dev/...` device names. The `dsinq(1)` command may be used to display the device names associated with a mounted volume.

The preferred way of accessing volumesets is with the commands `dsvaccess`, `dsvread`, `dsvwrite`, and `dsvrelease`. The commands `dsvmount` and `dsvunmount` provide an alternative way to access volumes for applications that need to interact directly with the device special files.

When accessing volumes with `dsvmount` and `dsvunmount`, there is no support for label processing, positioning, record blocking or End-of-Volume processing; DeepSpace assures that the proper volume is mounted.

Because the timeout period between `dsreserve` and `dsvmount` requests is short, it is necessary to issue the `dsvmount` command soon after the `dsreserve` command. To give yourself more time, include `idle=yes` in your `dsreserve` request.

24.4 Options

Argument	Valid Value
keywords	One or more keyword=value assignments. See Keywords[33]for a list of all valid keywords.
volset	Volumeset Name. Either the name given to the volumeset when it was submitted or period (`.`) followed by the volume ID of the first volume in the volumeset.
psd	Pseudo Device Name. An arbitrary name assigned during device reservation. This is used to distinguish between devices when multiple devices are reserved.

24.5 Keywords

Argument	Valid Value
`priority`	= Reservation Queue Priority. An integer value from 0 to 9 used to order reservation requests in the queue. Lower priority requests are serviced first. The default priority is 5.
`queue`	= Reservation Queue Flag. If set to yes and it cannot be satisfied immediately, the device reservation request is queued. The default is to return an error.
`write`	= Write Flag. By default, volumes are mounted read-only. Specifying write=yes allows both read and write access.
`passwd`	= Volume Access Password. An optional password which, if specified, must be given before the volumeset can be read or written. For example: `passwd=undertow` There is no specified default.
`force`	= Force attribute. Setting `force=yes` instructs the command to continue even if the volumeset is not expired. Only the owner of a volumeset can use the force keyword. The default is no.

24.6 See Also

dsreserve(1)(see page 53)
dsvunmount(1)(see page 105)

33 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=325353475#dsvmount(1)-KeywordSection

25 dsvmove(1)

25.1 Name

`dsvmove` — schedule a volumeset for movement to a different site.

25.2 Synopsis

`dsvmove location=`*`loc volset`*

On this page:

25.3 Description

`dsvmove` schedules all volumes in the named volset for movement to *loc*. *loc* must be a site known to DeepSpace. After the `dsvmove` command completes, the volumes in the volumeset will show up on the operator's Volume Maintenance Report as scheduled to move from the current location to the new location.

25.4 Options

Argument	Valid Value
`volset`	Volumeset Name. Either the name given to the volumeset when it was submitted or period (`.`) followed by the volume ID of the first volume in the volumeset.
`location`	= Volume Location. Location for the volume. The value given must be defined as an DeepSpace storage site. (The `dsreport --sites` command will produce a list of currently defined storage sites). There is no specified default.

25.5 See Also

dsreport(1)(see page 46)

26 dsvread(1)

26.1 Name

`dsvread` — read a file from a volumeset.

26.2 Synopsis

`dsvread [keywords] [psd]`

On this page:

26.3 Description

`dsvread` reads a file from the volumeset accessed (via `dsvaccess(1)`) on device *psd*. The file to read may be identified by its position in the volumeset with keyword `fseq` or its name with keyword `fid`. If `fseq` or `fid` are not given, the next file on the volumeset is read.

Output from `dsvread` can be directed to a file with keyword `of` or a FIFO with keyword `fifo`. By default, `dsvread` sends its output to `stdout`.

dsvread follows the UNIX convention of exiting with a zero return if successful and a non-zero on failure.

> ⓘ `dsvread` should not be interrupted. If the process is killed, use `dsvrelease` to release the referenced volumeset.

Options

Argument	Valid Value
keywords	One or more keyword=value assignments. See Keywords[34]for a list of all valid keywords.

34 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=325582859#dsvread(1)-KeywordSection

Argument	Valid Value
psd	Pseudo Device Name. An arbitrary name assigned during device reservation. This is used to distinguish between devices when multiple devices are reserved.

26.4 Keywords

Argument	Valid Value
`fifo`	= Output FIFO. If fifo is given, `dsvread` directs its output to the named FIFO. If the FIFO does not exist, it is created. When directing output to a FIFO, `dsvread` puts itself in the background after positioning to the proper file. The default is no.
`of`	= Output File. If `of` is given, `dsvread` directs its output to the named file. If the special value FID is given, output is directed to a file in the current directory whose name is the File Identifier of the file in the volumeset. If the special value SEQ is given, output is directed to a file in the current directory is the integer value of the file sequence number of the file in the volumeset. The default is `stdout`.
`fid`	= File ID. Specifies a tape file name. The assigned value has the general form: *fname*[:G*gno*][:V*vno*]. In this syntax, • *fname* is the file name, an arbitrary string up to 64 characters long. • G*gno* gives a file generation number. *gno* can be an integer or a signed integer. If *gno* is an integer, it references the given generation number. If *gno* is a signed integer, it references the highest existing generation number offset by *gno*. For example, if the highest generation number for file forecast is 10, :G-1 refers to generation 9 and :G+1 refers to generation 11. If the generation specification is omitted, the highest existing generation is referenced. If there are no existing generations, generation 0 is selected. The highest allowable generation number is 9999. You can use G+0 to find the latest (or most current) generation. • V*vno* gives a file version number. Version numbers behave like generation numbers. The highest allowable version number is 99. The default is the next available file. ⓘ Generation and version numbers are used as subscripts for files with the same vname.
`fseq`	= File Sequence Number. Selects a file by its relative position on the volumeset. Setting fseq=1 selects the first file, fseq=2 selects the second, etc. The default is the next available file

Argument	Valid Value
`fsect`	= File Section Number. Files that span multiple volumes are broken into sections. Normally, access starts at the first section (i.e. the beginning of the file). The fsect keyword may be used with either the fid or fseq keywords to initiate access at a specific section. The default is 1. ⓘ Spanned record formats (vs and vbs) should not be used with a file section number other than one.
`ortype`	= Output Record Type. Either fixed (default) or variable. The setting fixed has no effect on the output records. Specifying variable causes `dsvread` to prefix each output record with the record's length. The length is written as a six byte ASCII integer.
`span`	= Span Flag. By default `dsvread` and `dsvwrite` cross volume boundaries transparently. That is, an I/O operation that encounters End-of-Volume automatically requests the next volume. Setting span=no disables transparent volume crossing. In this case, the first End-of-Volume encountered is treated like End-of-File. The default is yes.

`rformat`	= Record Format. rformat has the general form: fmt:blen:rlen. In this syntax,
	
	• fmt is the record format:
	  • f - fixed length records.
	  • fb - fixed length, blocked records.
	  • v - variable length records.
	  • vb - variable length, blocked records.
	  • vs - variable length, spanned records.
	  • vbs - variable length, blocked, spanned records.
	  • u - unformatted data.
	• blen is the block length in bytes.
	• rlen is the record length in bytes.
	
	For example:
	
	`rformat=fb:800:80`
	
	There is no specified default.

`offset`	= Record offset. The number of bytes reserved at the beginning of each block for additional information. Offset only affects the ANSI tape format when the HDR2 labels are absent. The default is 0.
The following keywords are optional and an empty value is acceptable.	
`app`	= application. Only expired volumes belonging to the named application are scratched.

Argument	Valid Value
`conv`	= Record conversion specification. `conv` controls conversion of records to/from tape. `conv` may be set to one of the following values: • text for text record: • etext for EBCDIC text records. • data for fixed length ASCII or binary data records. • edata for fixed length EBCDIC data records.
`faccess`	= File Access Byte. The character assigned to faccess is written as byte 54 (Accessibility) in the HDR1 label. DeepSpace attaches no particular significance to the File Access byte; control is provided for export of volumes to sites that require certain values.
`fcomment`	= File comment. A comment about the file. If the comment includes spaces, it must be enclosed in quotes. It can be up to forty characters long. For example: `fcomment="my favorite file"`
`fexpire`	= File Expiration Date. It can be one of the following values: • I - infinite (never expires) • S - scratch (immediately expired) • RN - expires N days after creation • AN - expires if not accessed in N days. • L - expires when all files on the volumeset have expired. • Orotsched - follows the rotation schedule rotsched (see `dsrcreate(1)`). • Xccyymmdd or Xmm/dd/yy - expires on given date. • GN - expire when there are N newer generations. File expiration is only considered if volumeset expiration is set to L. For example: `cdate=R30` The default is S.
`force`	=Force attribute. Setting to y instructs the command to continue even if the volumeset is not expired. Only the owner of a volumeset can use the `force` keyword.
`fpasswd`	= File Access Password. An optional password which, if specified, must be given before the file can be read. For example: `fpasswd=undertow`

26.5 See Also

- dsvaccess(1)(see page 61)
- dsvrelease(1)(see page 87)

27 dsvrelease(1)

27.1 Name

`dsvrelease` — terminate access to a volumeset.

27.2 Synopsis

`dsvrelease [psd]`

On this page:

27.3 Description

`dsvrelease` releases a volumeset accessed with `dsvaccess(1)`. If the corresponding `dsvaccess` command reserved a device, the device is automatically freed.

27.4 Options

Argument	Valid Value
psd	Pseudo Device Name. An arbitrary name assigned during device reservation. This is used to distinguish between devices when multiple devices are reserved.

27.5 See Also

dsvread(1)(see page 82)

dsvwrite(1)(see page 106)

dsvaccess(1)(see page 61)

28 dsvretrieve(1)

28.1 Name

`dsvretrieve` — initiate volumeset retrieval.

28.2 Synopsis

`dsvretrieve` *`volset`*

On this page:

28.3 Description

`dsvretrieve` initiates retrieval of all volumes in a volumeset.

On completion, if DeepSpace is not configured to require volume acceptance, or, the volumes in *volset* have not yet been accepted, the volume records are deleted from the catalog, and retrieval is complete.

If DeepSpace does require volume acceptance, and, the volumes in the *volset* have been accepted, DeepSpace assigns each volume a receipt number. To complete retrieval, the receipt numbers are presented to an operator who completes the process and turns over the volumes.

28.4 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. See Keywords (see page 89)for a list of all valid keywords.
`volset`	Volumeset Name. Either the name given to the volumeset when it was submitted or '.' followed by the volume ID of the first volume in the volumeset.

28.5 Keywords

Argument	Valid Value
`prompt`	=If set to yes, prompts user to verify execution of `dsretrieve` process.

28.6 See Also

dsvsubmit(1)(see page 92)
dsvcreate(1)(see page 63)

29 dsvscratch(1)

29.1 Name

`dsvscratch` — scratch a volumeset.

29.2 Synopsis

`dsvscratch {[force={y|n}] volset | pool=pname | app=application}`

On this page:

29.3 Description

`dsvscratch` disbands *volset*. All constituent volumes are examined for cleaning or removal processing and then returned to scratch status in their pool. The volumeset owner can scratch a volumeset. If the volumeset has not expired, you must set force=y or `dsvscratch` will refuse to scratch it. The owner of the pool to which the constituent volumes belong may scratch a volumeset, but only if the volumeset has expired.

29.4 Options

Argument	Valid Value
keywords	One or more keyword=value assignments. See Keywords(see page 90) for a list of all valid keywords.
volset	Volumeset Name. Either the name given to the volumeset when it was submitted or a period (`.`) followed by the volume ID of the first volume in the volumeset.

29.5 Keywords

Argument	Valid Value
`force`	= Force attribute. Setting force=y instructs the command to continue even if the volumeset is not expired. Only the owner of a volumeset can use the force keyword. The default is no.
The following keywords are optional and an empty value is acceptable.	
`pool`	=All expired volumes in the named pool are scratched. If `pool=all` is used, then all pools are affected. However, default volumesets are not scratched. The full specification of pool is `userid/pool`. If `userid` is omitted, the current effective user ID is assumed. Only operators may create pools for IDs other than their own.
`app`	=application. Only expired volumes belonging to the named application are scratched.

29.6 See Also

dsvsubmit(1)(see page 92)
dsvcreate(1)(see page 63)

30 dsvsubmit(1)

30.1 Name

`dsvsubmit` — submit a volumeset.

30.2 Synopsis

`dsvsubmit` *`keywords`* [*`volset`*]

On this page:

30.3 Description

`dsvsubmit` initiates submission of a volumeset to DeepSpace. The `dsvsubmit` keywords describe attributes of the volumeset such as length, label format, media type, etc. If a keyword is not given, the corresponding attribute is given a default value. If the volumeset is a new Generation/Version, of an existing volumeset, default values are taken from the highest existing Generation/Version. Otherwise, DeepSpace looks for the closest matching Default Definition Record provided by the user. If the user has not provided a Default Definition Record, DeepSpace looks for Default Definition Records provided by root. (See the DeepSpace Operations Manual for a detailed description of Default Formats).

> The only keyword that is required is `lvsn`, which gives the volume serial numbers associated with the volumes belonging to the volumeset.

Upon successful completion, `dsvsubmit` displays the volume ID assigned to each constituent volume. The volume ID is a unique name for the volume within the catalog; it should be written on an external label affixed to the volume.

After `dsvsubmit` completes, the catalog contains a volume record for each volume in the volumeset. If DeepSpace is configured to require volume acceptance and/or volume identification, the volumeset is not accessible until it is handed over to an operator who uses `dsaccept(8)` and `dsid(8)` to complete submission.

30.4 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. There are no specified defaults associated with these keywords except where noted. See Keywords (see page 93)for a list of all valid keywords.
`volset`	Volumeset Name. A meaningful name assigned by the volumeset owner. The general form of *volset* is `[`*`userid/`*`]`*`vname`*`[:`*`Ggno`*`][:`*`Vvno`*`]`.In this syntax: • *userid* is a valid user ID. If userid is omitted, the effective user ID is assumed. Only an operator may submit to a *userid* other than the current effective user ID. • *vname* is an arbitrary string up to 12 characters long. • *Ggno* specifies a volumeset generation number. *gno* can be an integer or a signed integer. If *gno* is an integer, it references the given generation number. If *gno* is a signed integer, it references the highest existing generation number offset by *gno*. For example, if the highest generation number for volumeset test is 10, :G-1 refers to generation 9 and :G+1 refers to generation 11. If the generation specification is omitted, the highest existing generation is referenced. If there are no existing generations, generation 0 is selected. The highest allowable generation number is 9999. • *Vvno* specifies a volumeset version number. Generation and version numbers are used as subscripts for volumesets with the same *vname*. Version numbers behave like generation numbers. The highest allowable version number is 99. If *volset* is omitted, an unnamed volumeset is created. Unnamed volumesets are referenced by a period (`.`) followed by the volume ID of the first volume in the volumeset. For example, If the first volume has volume ID lump-992 the volumeset is referenced as `.lump-992`.

30.5 Keywords

Argument	Valid Value
`app`	= Application prefix. This prefix is used with file reports to limit selections to those with the specified `app`= value. app may be up to 12 characters long.
`capacity`	= Volume capacity in Mbytes. Specifies the volume capacity for cartridge media types.

Argument	Valid Value
`convert`	= Record conversion specification. convert controls conversion of records to/from tape. convert may be set to one of the following values: • text for text record: • etext for EBCDIC text records. • data for fixed length ASCII or binary data records. • edata for fixed length EBCDIC data records.
`dispose`	= Volume disposition. dispose controls when volumes leave the volumeset and what happens when they do. dispose may be set to erase and/or retain. If set to retain, the volumeset is never truncated, and all original volumes remain as volumeset members. If set to erase, whenever a file on the volumeset is overwritten, all volumes following the new file are scratched. To select both, use the following setting, including the quotation marks: `dispose="erase&retain"`
`ecnt`	= Error count. Set error count as specified.
`finger`	= File fingerprint. Set fingerprint field as specified.
`flocation`	= Free volume location. When a volumeset is truncated or scratched, the newly unattached volumes will return to the location assigned to flocation. The value given must be defined as a DeepSpace storage site (the `dsreport --sites` command will produce a list of currently defined storage sites). The value can be up to twelve characters long.
`format`	= Label Format. format may be set to one of the following values: • ANSI • IBM • IBMU • TAR • CPIO • RAW
`ftrack`	= File Tracking Flag. If set to yes, this causes the catalog to maintain a record for every file written to the volumeset. If set to no, this disables file cataloging. Maintaining file catalogs makes it possible to generate a volumeset table of contents without having to mount a tape. It also improves the efficiency of accessing files on multi-volume volumesets.

Argument	Valid Value

`lvsn`	= Volume Serial Number List. A comma separated list giving the Volume Serial Number (VSN) for each volume in the volumeset. *This keyword is required.*
	
	If the VSN is specified as agen, DeepSpace will automatically assign the volume a unique VSN. (In general, VSNs do not have to be unique.)
	
	In place of the comma separator, you may use `/` (slash) as a separator to indicate volumes that belong to the volumeset but do not yet contain data. For example, the following setting describes a volumeset with three volumes. The first two volumes contain existing files. The third volume is a spare to allow room for future expansion. The VSN for the third volume is automatically generated.
	
	`lvsn=000000,000001/agen`
	
	The `/` separator may occur at the beginning of the list to indicate a volumeset with no existing files. For example,
	
	`lvsn=/000000,000001`
	
	The values may be up to six characters long for each. The special value `vms` means that this value is selected by the Vault Management System.

`offset`	= Record offset. The number of bytes reserved at the beginning of each block for additional information. Offset only affects the ANSI tape format.

`passwd`	= Volume Access Password. An optional password which, if specified, must be given before the volumeset can be read or written. It can be up to fourteen characters long. For example:
	
	`password=undertow`
	
	`There is no specified default.`

`pool`	= Pool membership. Every volume must belong to a pool. By default, volumes belong to the user's private pool. Volumes may only be assigned to pools owned by the current effective user ID. This value may be up to twelve characters long. The full specification of pool is *userid*/*pool*. If *userid* is omitted, the current effective user ID is assumed. Only operators may create pools for IDs other than their own.

Argument	Valid Value

rformat	= Record Format. `rformat` has the general form: *`fmt:blen:rlen`*. In this syntax,
	
	• *fmt* is the record format:
	  • f - fixed length records.
	  • fb - fixed length, blocked records.
	  • v - variable length records.
	  • vb - variable length, blocked records.
	  • vs - variable length, spanned records.
	  • vbs - variable length, blocked, spanned records.
	  • u - unformatted data.
	• *blen* is the block length in bytes.
	• *rlen* is the record length in bytes.
	
	For example:
	
	`rformat=fb:800:80`
	
	The default is u.

`slocation`	= Scheduled Volume Location. If a volume's current location (`clocation`) does not equal its scheduled location, the volume will show up on the Volume maintenance report as wanting to move from `clocation` to `slocation`. The value given must be defined as an DeepSpace storage site. (The `dsreport --sites` command will produce a list of currently defined storage sites.) It can be up to twelve characters long. The default is the value of `clocation`.
`type`	= Volume Media Type. Media type is an arbitrary name assigned during DeepSpace configuration to describe volume formats supported by devices under DeepSpace control. For 9-track devices, this is typically the tape density (i.e. 1600, 6250, etc.). For cartridge devices, it is typically the cartridge standard (i.e. QIC11, QIC24, etc.). The given value must be a defined media type. The `dsreport --types` command produces a list of currently defined media types. It can be up to eight characters long.
`vaccess`	= Volume Access Byte. The character assigned to vaccess is written as byte 11 (Volume Accessibility) in the VOL1 label. DeepSpace attaches no particular significance to the Volume Access byte. Control is provided for export of volumes to sites that require certain values.

Argument	Valid Value
`vcomment`	= Volumeset comment. A comment about the volumeset. If the comment includes spaces, it must be enclosed in quotes. It can be up to forty characters long. For example: `vcomment="my favorite volumeset"`
`vexpire`	= Volumeset Expiration Date. It can be one of the following values: • I - infinite (never expires) • S - scratch (immediately expired) • RN - expires N days after creation • AN - expires if not accessed in N days. • L - expires when all files on the volumeset have expired. • Orotsched - follows the rotation schedule rotsched (see `dsrcreate(1)`) . • Xccyymmdd or Xmm/dd/yy - expires on given date. • GN - expire when there are N newer generations. For example: `vexpire=R30` The default is S. ⓘ Upon expiration, the volumeset is not disbanded until it is scratched. The `dsvscratch` program disbands volumesets.
`vmode`	= Volumeset Permission Mask. Three octal digits controlling owner, group and others (respectively) permission to the volumeset (similar to the UNIX file mode). Bit 4 controls read access. Bit 2 controls write access. Bit 1 permits viewing of the volume's catalog entry. For example: `vmode=744` The default is 700.
The following keywords may only be used by DeepSpace operators.	
`clean`	= Volume cleaning count. The number of times the volume has been mounted since it was last cleaned. This value is normally maintained by DeepSpace.
`clocation`	= Current Volume Location. The value given must be defined as an DeepSpace storage site. (The `dsreport --sites` command produces a list of currently defined storage sites. This value can be up to twelve characters long.

Argument	Valid Value
`ctype`	= Current Media Type. For initialized volumes, this field contains the Media Type DeepSpace has verified is the correct media type for the volume. This field is normally maintained by DeepSpace.
`group`	= Volumeset Group. The group (from /etc/group) to which the volumeset belongs. It can be up to twelve characters long. The default is the current effective group ID.
`maintenance`	=Scheduled Maintenance. Maintenance may be set to one or more of the following values: • move - indicates the volume is awaiting movement. • age - causes the volume to wait N days after it is scratched before becoming free (N is defined by the pool the tape belongs to). • erase - schedules the volume for erasure. • remove • clean - schedules the volume for cleaning. Multiple items are selected by separating them with `&` (ampersand). For example: `maint="erase&clean"` The default is no maintenance. Normally the maintenance value is maintained by DeepSpace.
`remove`	= Volume Usage Count. The number of times the volume has been mounted since it entered the catalog. This value is normally maintained by DeepSpace.
`status`	= Volume submission status. Status may be assigned the following integer values. • 1 - submitted awaiting acceptance. 2 - accepted awaiting identification. 3 - fully submitted. 4 - retrieved awaiting return. Submission status is normally maintained by DeepSpace.
`user`	= Volumeset Owner. The user ID (from `/etc/passwd`) to which the volumeset belongs. It can be up to twelve characters long. The default is the current effective user ID.
The following keywords are optional and an empty value is acceptable.	

Argument	Valid Value

`ftemplate`	= File Name template. The filename template is used to dynamically construct names for files written to the volumeset. The name constructed by the template is limited to seventeen (17) characters. The template consists of constant text and substitution patterns. Substitution patterns have the general form:`@sublen@`. In this syntax,
	
	• @ delimits the beginning and end of the substitution specification.
	• *`sub`* is a character indicating what to substitute.
	• *`len`* gives the length in characters of the substituted value. In principle it can be any number between 1 and 17. However, for some values of *sub* only a particular value of *len* is reasonable. The following values are recognized.
	  • @Y4@ – numeric year.
	  • @C2@ – numeric month (Jan = 1).
	  • @E3@ – month name.
	  • @D2@ – day of month.
	  • @J3@ – Julian day.
	  • @W3@ – day of week (Sun = 1).
	  • @H2@ – hour (24 hour clock).
	  • @M2@ – minute.
	  • @S2@ – second.
	  • @F9@ – value assigned fid keyword on the `dsvwrite` command.
	  • @U9@ – base name of file assigned to the if keyword on the `dsvwrite` command.
	  • @G4@ – file generation.
	  • @V2@ – file version.
	
	Numeric values are truncated on the left. Character values are truncated on the right. If a numeric value is shorter then len it is padded on the left with zeros. For example, on January 25 1987: ftemplate=GV, results in the following file name:
	
	`25Jan89G0000V01`
	
	ⓘ Even unlabeled volume formats (IBMU, RAW, TAR and CPIO) must be assigned volume serial numbers.

Argument	Valid Value
`lvid`	= Volume ID List. A comma separated list giving the volume IDs corresponding to the Volume Serial Number List (keyword lvsn). Volume IDs must be unique. Sites that maintain unique VSNs may make the volume ID the same as the VSN. By default, DeepSpace, assigns each volume a unique volume ID by combining a four character word with a three digit number. For example: `root-496` Each volume ID may be up to twelve characters long. The special value vms means that this value is selected by the Vault Management System.

30.6 See Also

dsreport(1)(see page 46)
dsvretrieve(1)(see page 88)

31 dsvtran(1)

31.1 Name

`dsvtran` — reinitialize a transient volumeset.

31.2 Synopsis

`dsvtran write={yes|no} volset`

On this page:

31.3 Description

A transient volumeset is a volumeset whose constituent volumes change periodically. For example, if a site receives weekly sales updates from a regional sales office on tape, it would be reasonable to create a transient volumeset 'sales_up'. Each week, referring to volumeset 'sales_up' selects the latest sales information.

There is no real difference between a normal volumeset and a volumeset that is considered transient. The only trick is when a mount request is made for a transient volumeset, the operator must be able to recognize that the requested volume will not be found in the tape library, but in some other location (probably the incoming mail). This is usually accomplished by giving the rack number a special value. In this example, the rack number could be 'sales_up'. If it is a multi-volume volumeset the volumes might have rack numbers 'sales_up.1', 'sales_up.2', etc.

The difficulty for DeepSpace in managing a transient volumeset is detailed information about the constituent volumes changes periodically. For example, the volume fingerprint that is taken the first time the volumeset is mounted will be incorrect when a new volume arrives. The command `dsvtran` resets the necessary volume information so a new transient volume can be accessed successfully. Specifically, `dsvtran` wipes out all associated file records, sets the volume fingerprint to unknown, the volume type to unknown, and if write=yes is specified, turns on volume initialization, otherwise it sets the volume serial number to unknown. The `dsvtran` command should be given before accessing a transient volumeset with new constituent volumes.

31.4 Options

Argument	Valid Value
keywords	One or more keyword=value assignments. See Keywords (see page 102)for a list of all valid keywords.
volset	Volumeset Name. Either the name given to the volumeset when it was submitted or a period (`.`) followed by the volume ID of the first volume in the volumeset.

31.5 Keywords

Argument	Valid Value
`write`	= yes | no. The default is no.

31.6 See Also

- dsvsubmit(1)(see page 92)
- dsvcreate(1)(see page 63)

32 dsvtruncate(1)

32.1 Name

`dsvtruncate` — truncate a volumeset.

32.2 Synopsis

`dsvtruncate fseq=`*`fileno volset`*

On this page:

32.3 Description

`dsvtruncate` truncates the named volumeset at the specified file sequence number. For example, if a volumeset consists of ten files on eight tapes and the tenth file alone consumes the last three tapes, then this command removes the last three tapes from the volumeset when the file sequence number '10' is specified. The other nine files are left unaltered on the remaining five tapes. The three truncated tapes are expired.

32.4 Options

Argument	Valid Value
`keywords`	One or more keyword=value assignments. See Keywords (see page 103)for a list of all valid keywords.
`volset`	Volumeset Name. Either the name given to the volumeset when it was submitted or a period (`.`) followed by the volume ID of the first volume in the volumeset.

32.5 Keywords

Argument	Valid Value
`fseq`	=The file sequence number of the file targeted. Volumeset files are numbered starting with 1.

32.6 See Also

dsreport(1)(see page 46)

33 dsvunmount(1)

33.1 Name

`dsvunmount` — unmount a volume.

33.2 Synopsis

`dsvunmount [psd]`

On this page:

33.3 Description

`dsvunmount` unmounts the volume associated with device *psd*. If the corresponding `dsvmount` command reserved the device, the device is automatically freed.

33.4 Options

Argument	Valid Value
psd	Pseudo Device Name. An arbitrary name assigned during device reservation. This is used to distinguish between devices when multiple devices are reserved. If *psd* is not specified, all devices under the user's control are freed.

33.5 See Also

dsvmount(1)(see page 79)

34 dsvwrite(1)

34.1 Name

`dsvwrite` — write a file to a volumeset.

34.2 Synopsis

`dsvwrite` *`keywords`* [*`psd`*]

On this page:

34.3 Description

`dsvwrite` writes a file to the volumeset accessed (via `dsvaccess(1)`) on device *psd*. By default, `dsvwrite` writes the new file at the current volumeset position. The `fseq` keyword may be used to write the file at a location other than the current location.

Input to `dsvwrite` can be taken from a file with keyword `if` or a FIFO with keyword `fifo`. By default, `dsvwrite` gets its input from `stdin`.

`dsvwrite` follows the UNIX convention of exiting with a zero return if successful and a non-zero on failure.

> ! `dsvwrite` should not be interrupted. If the process is killed, use `dsvrelease` to release the referenced volumeset.

34.4 Options

Argument	Valid Value
keywords	One or more keyword=value assignments. See Keywords[35]for a list of all valid keywords.
psd	Pseudo Device Name. An arbitrary name assigned during device reservation. This is used to distinguish between devices when multiple devices are reserved.

34.5 Keywords

Argument	Valid Value
`app`	=Application prefix. This prefix is used with file reports to limit selections to those with the specified `app=` value. app may be up to 12 characters long. There is no specified default.
`if`	=Input File. If the `if` keyword is given, `dsvwrite` gets its input from the named file. The default is `stdin`.
`fid`	= File ID. Specifies a tape file name. The assigned value has the general form: *`fname`*`[:`*`Ggno`*`][:`*`Vvno`*`]`. In this syntax, • *fname* is the file name, an arbitrary string up to 256 characters long. • *Ggno* gives a file generation number. *gno* can be an integer or a signed integer. If *gno* is an integer, it references the given generation number. If *gno* is a signed integer, it references the highest existing generation number offset by *gno*. For example, if the highest generation number for file forecast is 10, :G-1 refers to generation 9 and :G+1 refers to generation 11. If the generation specification is omitted, the highest existing generation is referenced. If there are no existing generations, generation 0 is selected. The highest allowable generation number is 9999. You can use G+0 to find the latest (or most current) generation. • *Vvno* gives a file version number. Version numbers behave like generation numbers. The highest allowable version number is 99. The default is the next available file.
`fseq`	= File Sequence Number. Selects a file by its relative position on the volumeset. Setting fseq=1 selects the first file, fseq=2 selects the second, etc. fseq=EOT appends to the volumeset without specifying a file sequence number. The default is the next available file

35 https://deepspace.atlassian.net/wiki/pages/resumedraft.action?draftId=325910531#dsvwrite(1)-KeywordSection

Argument	Valid Value
`fsect`	= File Section Number. Files that span multiple volumes are broken into sections. Normally, access starts at the first section (i.e. the beginning of the file). The `fsect` keyword may be used with either the `fid` or `fseq` keywords to initiate access at a specific section. The default is 1. ⓘ Spanned record formats (vs and vbs) should not be used with a file section number other than one.
`irtype`	= Input Record Type. Can be set to either fixed (the default setting) or variable. When set to variable, it looks for a six byte ASCII integer at the beginning of each input line. When set to fixed, `dsvwrite` does not look for the prefixed integer.
`span`	= Span Flag. By default `dsvread` and `dsvwrite` cross volume boundaries transparently. That is, an I/O operation that encounters End-of-Volume automatically requests the next volume. Setting `span=no` disables transparent volume crossing. In this case, the first End-of-Volume encountered is treated like End-of-File. The default is yes.

`rformat`	= Record Format. rformat has the general form: *`fmt:blen:rlen`*. In this syntax,
	
	• *fmt* is the record format:
	  • f - fixed length records.
	  • fb - fixed length, blocked records.
	  • v - variable length records.
	  • vb - variable length, blocked records.
	  • vs - variable length, spanned records.
	  • vbs - variable length, blocked, spanned records.
	  • u - unformatted data.
	• *blen* is the block length in bytes.
	• *rlen* is the record length in bytes.
	
	For example:
	
	`rformat=fb:800:80`
	
	There is no specified default.

`offset`	= Record offset. The number of bytes reserved at the beginning of each block for additional information. The number of bytes reserved at the beginning of each block for additional information. Offset only affects the ANSI tape format. There is no specified default.

Argument	Valid Value
`convert`	= Record conversion specification. `convert` controls conversion of records to/from tape. `convert` may be set to one of the following values: • text for text record: • etext for EBCDIC text records. • data for fixed length ASCII or binary data records. • edata for fixed length EBCDIC data records. There is no specified default.
`faccess`	= File Access Byte. The character assigned to faccess is written as byte 54 (Accessibility) in the HDR1 label. DeepSpace attaches no particular significance to the File Access byte; control is provided for export of volumes to sites that require certain values. There is no specified default.
`fexpire`	= File Expiration Date. It can be one of the following values: • I - infinite (never expires) • S - scratch (immediately expired) • RN - expires N days after creation • AN - expires if not accessed in N days. • L - expires when all files on the volumeset have expired. • Orotsched - follows the rotation schedule rotsched (see `dsrcreate(1)`). • Xccyymmdd or Xmm/dd/yy - expires on given date. • GN - expire when there are N newer generations. File expiration is only considered if volumeset expiration is set to L. For example: `cdate=R30` The default is S.
`force`	=Force attribute. Setting to y instructs the command to continue even if the volumeset is not expired. Only the owner of a volumeset can use the force keyword. The default is no.
`fmode`	= File Permission Mask. Three octal digits controlling owner, group and others (respectively) permission to the file. Bit 4 controls read access. Bit 2 controls write access. Bit 1 permits viewing of the file's catalog entry. For example: `fmode=744` There is no specified default.
The following keywords are optional and an empty value is acceptable.	

Argument	Valid Value
`fcomment`	= File comment. A comment about the file. If the comment includes spaces, it must be enclosed in quotes. It can be up to 60 characters long. For example: `fcomment="my favorite file"`
`fifo`	= Input FIFO. If `fifo` is given, `dsvwrite` takes its input from the named FIFO. If the FIFO does not exist, it is created. When taking input from a FIFO, `dsvwrite` puts itself in the background after positioning to the proper file.
`fpasswd`	= File Access Password. An optional password which, if specified, must be given before the file can be read. It can be up to fourteen characters long.
`warn`	= Warning messages. If set to yes, warning messages will be provided. The default is no.

34.6 See Also

dsvread(1)(see page 82)
dsvaccess(1)(see page 61)