IBM System Storage SAN Volume Controller

Command-Line Interface User's Guide

Version 4.3.1

SC26-7903-04

IBM System Storage SAN Volume Controller

Command-Line Interface User's Guide

Version 4.3.1

SC26-7903-04

Note:
Before using this information and the product it supports, read the information in **Notices**.

This edition applies to the IBM System Storage SAN Volume Controller, release 4.3.1, and to all subsequent releases and modifications until otherwise indicated in new editions. This edition replaces SC26-7903-03 and all previous versions of SC26-7544.

Contents

Tables

About this guide

This publication provides information that helps you configure and use the IBM® System Storage™ SAN Volume Controller.

Who should use this guide

This guide is intended for system administrators or others who install and use the SAN Volume Controller.

Before you use the SAN Volume Controller, you should have an understanding of storage area networks (SANs), the storage requirements of your enterprise, and the capabilities of your storage units.

Summary of changes

This document contains terminology, maintenance, and editorial changes.

Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change. This summary of changes describes new functions that have been added to this release.

Summary of changes for SC26-7903-04 SAN Volume Controller Command-Line Interface User's Guide

This topic describes the changes that have been made to the SAN Volume Controller *Command-Line Interface User's Guide* since the previous edition (SC26-7903-03).

New information

The following new SAN Volume Controller commands have been added for edition SC26-7903-04:

- Information commands:
  - lscimomdumps
- Service-mode information commands:
  - lscimomdumps
- Virtual disk commands:
  - recovervdisk
  - recovervdiskbycluster
  - recovervdiskbyiogrp

Changed information

The following commands have been updated for edition SC26-7903-04:

- addmdisk
- addvdiskcopy
- backup
- chcluster
- chcontroller

- chemail
- chfcmap
- chlicense
- chpartnership
- chvdisk
- clear
- cleardumps
- cpdumps
- lscluster
- lscontroller
- lsfabric
- lsiogrp
- lslicense
- lsnode
- lsnodevpd
- lsvdisk
- lsvdiskcopy
- migrateexts
- migratetoimage
- migratevdisk
- mkfcmap
- mkvdisk
- repairvdiskcopy
- rmmdisk
- rmnode
- rmpartnership
- rmvdisk
- setemail
- setevent

Summary of changes for SC26-7903-03 SAN Volume Controller Command-Line Interface User's Guide

This topic describes the changes that have been made to the SAN Volume Controller *Command-Line Interface User's Guide* since the previous edition (SC26-7903-02).

New information

The following new SAN Volume Controller commands have been added for edition SC26-7903-03:

- Information commands:
  - lscontrollerdependentvdisks
  - lsmdisklba
  - lsrepairsevdiskcopyprogress
  - lsrepairvdiskcopyprogress
  - lssevdiskcopy

  - lssoftwareupgradestatus
  - lsvdiskcopy
  - lsvdisklba
  - lsvdisksyncprogress
- Virtual disk commands:
  - addvdiskcopy
  - repairsevdiskcopy
  - repairvdiskcopy
  - rmvdiskcopy
  - splitvdiskcopy

Changed information

The following commands have been updated for edition SC26-7903-03:

- addmdisk
- caterrlog
- caterrlogbyseqnum
- chcluster
- chemail
- chemailuser
- chiogrp
- chfcconsistgrp
- chmdiskgrp
- chvdisk
- cleardumps
- expandvdisksize
- ls2145dumps
- lsauditlog
- lsauth
- lscluster
- lsclustercandidate
- lscontroller
- lscopystatus
- lsdiscoverystatus
- lserrlogbyfcconsistgrp
- lserrlogbyfcmap
- lserrlogbyhost
- lserrlogbyiogrp
- lserrlogbymdisk
- lserrlogbymdiskgrp
- lserrlogbynode
- lserrlogbyrcconsistgrp
- lserrlogbyrcrelationship
- lserrlogbyvdisk
- lserrlogdumps
- lsfabric

- lsfcconsistgrp
- lsfcmap
- lsfcmapcandidate
- lsfcmapprogress
- lsfcmapdependentmaps
- lsfeaturedumps
- lsfreeextents
- lshbaportcandidate
- lshost
- lshostiogrp
- lshostvdiskmap
- lsiogrp
- lsiogrphost
- lsiogroupcandidate
- lsiostatsdumps
- lsiotracedumps
- lslicense
- lsmdisk
- lsmdisklba
- lsmdiskcandidate
- lsmdiskextent
- lsmdiskgrp
- lsmdiskmember
- lsmigrate
- lsnode
- lsnodecandidate
- lsnodevpd
- lsrcconsistgrp
- lsrcrelationship
- lsrcrelationshipcandidate
- lsrcrelationshipprogress
- lssoftwaredumps
- lssoftwareupgradestatus
- lssshkeys
- lstimezones
- lsvdisk
- lsvdiskdependentmaps
- lsvdiskextent
- lsvdiskfcmappings
- lsvdiskhostmap
- lsvdisklba
- lsvdiskmember
- lsvdiskprogress
- migrateextents
- migratetoimage

- migratevdisk
- mkemailuser
- mkfcconsistgrp
- mkfcmap
- mkmdiskgrp
- mkvdisk
- rmemailuser
- rmmdisk
- rmmdiskgrp
- setevent
- setquorum
- showtimezone
- shrinkvdisksize
- testemail
- setemail
- setlocale
- setquorum
- shrinkvdisksize

Emphasis

Different typefaces are used in this guide to show emphasis.

The following typefaces are used to show emphasis:

Boldface	Text in **boldface** represents menu items and command names.
Italics	Text in *italics* is used to emphasize a word. In command syntax, it is used for variables for which you supply actual values, such as a default directory or the name of a cluster.
`Monospace`	Text in `monospace` identifies the data or commands that you type, samples of command output, examples of program code or messages from the system, or names of command flags, parameters, arguments, and name-value pairs.

SAN Volume Controller library and related publications

A list of other publications that are related to this product are provided to you for your reference.

The tables in this section list and describe the following publications:

- The publications that make up the library for the IBM System Storage SAN Volume Controller
- Other IBM publications that relate to the SAN Volume Controller

SAN Volume Controller library

The following table lists and describes the publications that make up the SAN Volume Controller library. Unless otherwise noted, these publications are available in Adobe® portable document format (PDF) from the following Web site:

www.ibm.com/storage/support/2145

Title	Description	Order number
IBM System Storage SAN Volume Controller Planning Guide	This guide introduces the SAN Volume Controller and lists the features you can order. It also provides guidelines for planning the installation and configuration of the SAN Volume Controller.	GA32-0551
IBM System Storage SAN Volume Controller Model 2145-8A4 Hardware Installation Guide	This guide provides the instructions that the IBM service representative uses to install the hardware for SAN Volume Controller model 2145-8A4.	GC27-2219
IBM System Storage SAN Volume Controller Model 2145-8G4 Hardware Installation Guide	This guide provides the instructions that the IBM service representative uses to install the hardware for SAN Volume Controller model 2145-8G4.	GC27-2220
IBM System Storage SAN Volume Controller Software Installation and Configuration Guide	This guide provides guidelines for configuring your SAN Volume Controller. Instructions for backing up and restoring the cluster configuration, using and upgrading the SAN Volume Controller Console, using the CLI, upgrading the SAN Volume Controller software, and replacing or adding nodes to a cluster are included.	SC23-6628
IBM System Storage SAN Volume Controller CIM Agent Developer's Guide	This guide describes the concepts of the Common Information Model (CIM) environment. Steps about using the CIM agent object class instances to complete basic storage configuration tasks, establishing new Copy Services relationships, and performing CIM agent maintenance and diagnostic tasks are included.	SC23-6665

Title	Description	Order number
IBM System Storage SAN Volume Controller Command-Line Interface User's Guide	This guide describes the commands that you can use from the SAN Volume Controller command-line interface (CLI).	SC26-7903
IBM System Storage SAN Volume Controller Host Attachment Guide	This guide provides guidelines for attaching the SAN Volume Controller to your host system.	SC26-7905
IBM System Storage SAN Volume Controller Troubleshooting Guide	This guide describes the features of each SAN Volume Controller model, explains how to use the front panel, and provides maintenance analysis procedures to help you diagnose and solve problems with the SAN Volume Controller.	GC27-2227
IBM System Storage SAN Volume Controller Hardware Maintenance Guide	This guide provides the instructions that the IBM service representative uses to service the SAN Volume Controller hardware, including the removal and replacement of parts.	GC27-2226
IBM System Storage SAN Volume Controller Models 2145-8F2 and 2145-8F4 Hardware Installation Guide	This guide provides the instructions that the IBM service representative uses to install the hardware for SAN Volume Controller models 2145-8F2 and 2145-8F4.	GC27-2221
IBM System Storage SAN Volume Controller Model 2145-4F2 Hardware Installation Guide	This guide provides the instructions that the IBM service representative uses to install the hardware for SAN Volume Controller model 2145-4F2.	GC27-2222
IBM System Storage SAN Volume Controller Master Console Guide	This guide describes how to install, maintain, and service the master console.	GC27-2223
IBM Systems Safety Notices	This guide contains translated caution and danger statements. Each caution and danger statement in the SAN Volume Controller documentation has a number that you can use to locate the corresponding statement in your language in the *IBM Systems Safety Notices* document.	G229-9054

Other IBM publications

The following table lists and describes other IBM publications that contain additional information that is related to the SAN Volume Controller.

You can download IBM eServer™ IBM xSeries® and IBM System x™ publications from the following Web site:

www.ibm.com/jct01004c/systems/support/

Title	Description	Order number
IBM System Storage Productivity Center Introduction and Planning Guide	This guide introduces the IBM System Storage Productivity Center hardware and software.	SC23-8824
Read This First: Installing the IBM System Storage Productivity Center	This guide describes how to install the IBM System Storage Productivity Center hardware.	GI11-8938
IBM System Storage Productivity Center User's Guide	This guide describes how to configure the IBM System Storage Productivity Center software.	SC27-2336
IBM System Storage Multipath Subsystem Device Driver User's Guide	This guide describes the IBM System Storage Multipath Subsystem Device Driver for IBM System Storage products and how to use it with the SAN Volume Controller.	GC52-1309
IBM System Storage DS Storage Manager Version 10.30 Installation and Host Support Guide	This guide provides information about how to plan, install, configure, and work with IBM System Storage DS™ Storage Manager.	GC53-1135
IBM System Storage DS Storage Manager Version 10.30 Copy Services Guide	This guide provides information about setting up, installing, configuring, and working with the three IBM System Storage DS Storage Manager premium features that assist with copy services: FlashCopy®, VolumeCopy, and the Enhanced Remote Mirroring Option.	GC53-1136
IBM System Storage DS4000/DS5000 Fibre Channel and Serial ATA Intermix Premium Feature Installation Overview	This overview describes how to install the IBM System Storage DS4000/DS5000 Fibre Channel and Serial ATA Intermix Premium Feature.	GC53-1137
IBM System Storage DS5100 and DS5300 Installation, User's and Maintenance Guide	This guide describes how to install and configure the IBM System Storage DS5100 and DS5300.	GC53-1140

Title	Description	Order number
IBM System Storage EXP5000 Storage Expansion Enclosure Installation, User's, and Maintenance Guide	This guide describes how to install and configure the IBM System Storage EXP5000 Storage Expansion Enclosure.	GC53-1141
IBM System Storage DS Storage Manager Command-line Programming Guide	This guide describes the commands that you can use from the IBM System Storage DS Storage Manager command-line interface.	GC52-1275
IBM System Storage DS5000 Quick Start Guide: Quick Reference for the DS5100, DS5300 and EXP5000	This guide provides information about setting up and installing the DS5100, DS5300 and EXP5000.	GC53-1134
IBM TotalStorage DS4300 Fibre Channel Storage Subsystem Installation, User's, and Maintenance Guide	This guide describes how to install and configure the IBM TotalStorage® DS4300 Fibre-Channel Storage Subsystem.	GC26-7722
IBM eServer xSeries 306m (Types 8849 and 8491) Installation Guide	This guide describes how to install the IBM eServer xSeries 306m, which is the hardware delivered for some versions of the hardware master console.	MIGR-61615
IBM xSeries 306m (Types 8849 and 8491) User's Guide	This guide describes how to use the IBM eServer xSeries 306m, which is the hardware delivered for some versions of the hardware master console.	MIGR-61901
IBM xSeries 306m (Types 8849 and 8491) Problem Determination and Service Guide	This guide can help you troubleshoot and resolve problems with the IBM eServer xSeries 306m, which is the hardware delivered for some versions of the hardware master console.	MIGR-62594
IBM eServer xSeries 306 (Type 8836) Installation Guide	This guide describes how to install the IBM eServer xSeries 306, which is the hardware delivered for some versions of the hardware master console.	MIGR-55080
IBM eServer xSeries 306 (Type 8836) User's Guide	This guide describes how to use the IBM eServer xSeries 306, which is the hardware delivered for some versions of the hardware master console.	MIGR-55079

Title	Description	Order number
IBM eServer xSeries 306 (Types 1878, 8489 and 8836) Hardware Maintenance Manual and Troubleshooting Guide	This guide can help you troubleshoot problems and maintain the IBM eServer xSeries 306, which is the hardware delivered for some versions of the hardware master console.	MIGR-54820
IBM eServer xSeries 305 (Type 8673) Installation Guide	This guide describes how to install the IBM eServer xSeries 305, which is the hardware delivered for some versions of the hardware master console.	MIGR-44200
IBM eServer xSeries 305 (Type 8673) User's Guide	This guide describes how to use the IBM eServer xSeries 305, which is the hardware delivered for some versions of the hardware master console.	MIGR-44199
IBM eServer xSeries 305 (Type 8673) Hardware Maintenance Manual and Troubleshooting Guide	This guide can help you troubleshoot problems and maintain the IBM eServer xSeries 305, which is the hardware delivered for some versions of the hardware master console.	MIGR-44094
IBM TotalStorage 3534 Model F08 SAN Fibre Channel Switch User's Guide	This guide introduces the IBM TotalStorage SAN Switch 3534 Model F08.	GC26-7454
IBM System x3250 (Types 4364 and 4365) Installation Guide	This guide describes how to install the IBM System x3250, which is the hardware delivered for some versions of the hardware master console.	MIGR-5069761
IBM System x3250 (Types 4364 and 4365) User's Guide	This guide describes how to use the IBM System x3250, which is the hardware delivered for some versions of the hardware master console.	MIGR-66373
IBM System x3250 (Types 4364 and 4365) Problem Determination and Service Guide	This guide can help you troubleshoot and resolve problems with the IBM System x3250, which is the hardware delivered for some versions of the hardware master console.	MIGR-66374
IBM TotalStorage SAN Switch 2109 Model F16 User's Guide	This guide introduces the IBM TotalStorage SAN Switch 2109 Model F16.	GC26-7439

Title	Description	Order number
IBM TotalStorage SAN Switch 2109 Model F32 User's Guide	This guide introduces the IBM TotalStorage SAN Switch 2109 Model F32. It also describes the features of the switch and tells you where to find more information about those features.	GC26-7517

Some related publications are available from the following SAN Volume Controller support Web site:

www.ibm.com/storage/support/2145

Related Web sites

The following Web sites provide information about the SAN Volume Controller or related products or technologies:

Type of information	Web site
SAN Volume Controller support	www.ibm.com/storage/support/2145
Technical support for IBM storage products	www.ibm.com/storage/support/

How to order IBM publications

The IBM Publications Center is a worldwide central repository for IBM product publications and marketing material.

The IBM Publications Center offers customized search functions to help you find the publications that you need. Some publications are available for you to view or download at no charge. You can also order publications. The publications center displays prices in your local currency. You can access the IBM Publications Center through the following Web site:

www.ibm.com/shop/publications/order/

How to send your comments

Your feedback is important to help us provide the highest quality information. If you have any comments about this book or any other documentation, you can submit them in one of the following ways:

- E-mail

  Submit your comments electronically to the following e-mail address:

  starpubs@us.ibm.com

  Be sure to include the name and order number of the book and, if applicable, the specific location of the text you are commenting on, such as a page number or table number.
- Mail

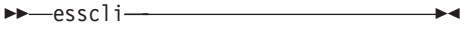
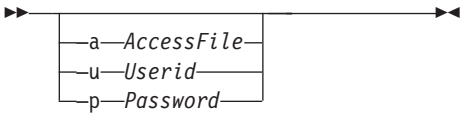
Fill out the Readers' Comments form (RCF) at the back of this book. If the RCF has been removed, you can address your comments to:

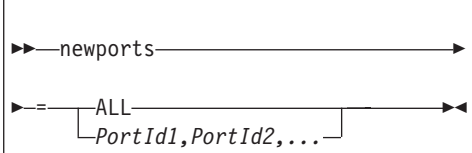
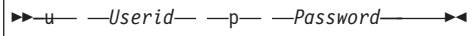
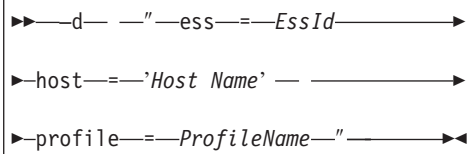
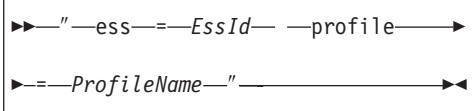
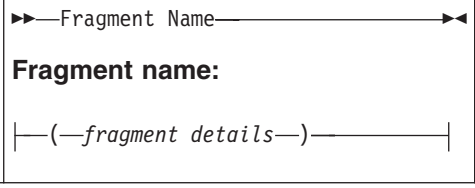
International Business Machines Corporation
RCF Processing Department
Department 61C
9032 South Rita Road
Tucson, Arizona 85775-4401
U.S.A.

Syntax diagrams

A syntax diagram uses symbols to represent the elements of a command and to specify the rules for using these elements.

This explains how to read the syntax diagrams that represent the command-line interface (CLI) commands. In doing so, it defines the symbols that represent the CLI command elements.

Element	Syntax	Description
Main path line	>>><>() () ()	>>Begins on the left with double arrowheads ()>< and ends on the right with two arrowheads facing each other (). If a diagram is longer than one line, each line to be continued ends with a single> arrowhead () and the next line begins with a single arrowhead. Read the diagrams from left–to–right, top–to–bottom, following the main path line.
Keyword		Represents the name of a command, flag, parameter, or argument. A keyword is not in italics. Spell a keyword exactly as it is shown in the syntax diagram.
Required keywords		Indicate the parameters or arguments you must specify for the command. Required keywords appear on the main path line. Required keywords that cannot be used together are stacked vertically.
Optional keywords		Indicate the parameters or arguments you can choose to specify for the command. Optional keywords appear below the main path line. Mutually exclusive optional keywords are stacked vertically.

Element	Syntax	Description
Default value		Appears above the main path line.
Repeatable keyword or value		Represents a parameter or argument that you can specify more than once. A repeatable keyword or value is represented by an arrow returning to the left above the keyword or value.
Variable		Represents the value that you need to supply for a parameter or argument, such as a file name, user name, or password. Variables are in italics.
Space separator		Adds a blank space on the main path line to separate keywords, parameters, arguments, or variables from each other.
Quotation mark delimiters		Indicates the start and end of a parameter or argument that contains multiple values. Enclose one or more name–value pairs in a set of double quotation marks for a particular parameter or argument. If the value of a parameter or name–value pair contains a blank or white space, enclose the entire value in a set of single quotation marks.
Equal–sign operator		Separates a name from its value in a name–value pair.
Syntax fragment		Breaks up syntax diagrams that are too long, too complex, or repetitious. The fragment name is inserted in the main diagram, and the actual fragment is shown below the main diagram.

Terminology

These are abbreviations that are most commonly used for the command-line interface operations.

The following table shows the abbreviations that are most commonly used for the command-line interface operations.

Name	Object type
Host	host
Virtual disk	vdisk
Virtual disk copy	vdiskcopy
Space-efficient virtual disk copy	sevdiskcopy
Managed disk	mdisk
Managed disk group	mdiskgrp
I/O group	iogrp
Node	node
Cluster	cluster
Controller	controller
IBM FlashCopy® mapping	fcmap
FlashCopy consistency group	fcconsistgrp
Metro Mirror or Global Mirror relationship	rcrelationship
Metro Mirror or Global Mirror consistency group	rcconsistgrp
Unsupported/unknown object	unknown

CLI special characters

The following special characters are used in the command-line interface (CLI) command examples.

minus (-) sign

Flags are prefixed with a - (minus) sign. Flags define the action of a command or modify the operation of a command. You can use multiple flags, followed by parameters, when you issue a command. The - character cannot be used as the first character of an object name.

vertical bar (|)

A vertical bar signifies that you choose only one value. For example, `[ a | b ]` in brackets indicates that you can choose a, b, or nothing. Similarly, `{ a | b }` in braces indicates that you must choose either a or b.

Using wildcards in the SAN Volume Controller CLI

You can use wildcards in the SAN Volume Controller Command-Line Interface.

The SAN Volume Controller supports the use of the asterisk character (*) as a wildcard within the arguments of certain parameters. There are some behavioral issues that must be considered when using wildcards in order to prevent unexpected results. These behavioral issues and the ways to avoid them are as follows:

1. Running the command while logged onto the node.

   The shell will attempt to interpret any of the special characters if they are not escaped (preceded with a backslash character). Wildcards will be expanded into a list of files if any files exist that match the wildcards. If no matching files exist, the wildcard is passed to the SAN Volume Controller command untouched.

   To prevent expansion, issue the following command in one of its formats:

svctask cleardumps -prefix '/dumps/*.txt' with single quotation marks (''), or

svctask cleardumps -prefix /dumps/*.txt using a backslash (\), or

svctask cleardumps -prefix "/dumps/*.txt" with double quotation marks ("").

2. Running the command through Secure Shell (SSH), for example from a host.

   This method is slightly more complicated because the host shell will process the command line before it is passed through SSH to the shell on the cluster. This means an extra layer of protection is required around the wildcard as the host shell will strip off any protecting quotes and if the wildcard is exposed to the cluster shell, then this will result in the wildcard being expanded in the cluster shell.

   To prevent expansion, issue the following command in one of its formats:

   svctask cleardumps "'/dumps/*.txt'" with single quotation marks ('') inside of double quotation marks (""), or

   svctask cleardumps '/dumps/*.txt' using a backslash (\) inside of single quotation marks (''), or

   svctask cleardumps '"/dumps/*.txt"' with double quotation marks ("") inside of single quotation marks ('').

Data types and value ranges

The maximum length of any single parameter that is entered into the command line is 2176 bytes.

Note: If you do not specify a name when you are creating a new object, the cluster assigns a default name. This name is made from the object type as the prefix and the object ID as the suffix. For example, a new virtual disk (VDisk) is created with ID 5. This object is given the default name of **vdisk5**. Because the system assigns these names, it does not allow you to create an object and call it **vdisk***x* where *x* is the integer. This is because the cluster reserves these names (for example, object_type_prefix integer) for default.

Data types	Value ranges
filename_arg	This is a (optionally fully qualified) file name. Maximum length is 169 characters. Valid characters consist of the following options: • . • / • - • _ • a - z • A - Z • 0 - 9 The field must not contain two consecutive '.', or start with a '.', or end with a '.'.

Data types	Value ranges
directory_or_file_filter	Specifies a directory and or file name filter within the specified directory. Valid directory values consist of the following options: • /dumps • /dumps/audit • /dumps/configs • /dumps/elogs • /dumps/feature • /dumps/iostats • /dumps/iotrace • /dumps/software The file name filter can be any valid file name with or without the wildcard '*'. The file name filter can be appended to the end of one of the previous directory values. Maximum length is 128 characters. Valid characters consist of the following options: • * • . • / • - • _ • a - z • A - Z • 0 - 9 The field must not contain two consecutive '.', or start with a '.', or end with a '.'.
filename_prefix	This is a prefix to be used when naming a file. Maximum length is 128 characters. Valid characters consist of the following options: • a - z • A - Z • 0 - 9 • - • _

Data types	Value ranges
name_arg	Names can be specified or changed using the create and modify functions. With the view commands you can see both the name and ID of an object. A string of 1 - 15 characters can be used, composed of characters A - Z, a - z, 0 - 9, - (dash), and _ (underscore). The first character of a *name_arg* must not be numeric. The first character of an object name cannot be a - (dash) because the CLI interprets it as being the next parameter. When creating a name for an object, the name must not consist of the object type followed only by an integer. The exception is Metro or Global Mirror relationships, which can be named anything as long as the names are unique across the two clusters. This naming convention is used by the system to generate default names. You cannot use one of the following reserved words followed by an integer: • cluster • controller • fccstgrp • fcmap • host • io_grp • mdisk • mdiskgrp • node • rccstgrp • rcmap The cluster name is set when the cluster is created.
password	This is a user-defined password. A password must meet the following requirements: • Can use a - z, A - Z, 0 - 9 in any sequence • Can use - (dash) but not as the first character • Can use _ (underscore) • Can contain a maximum of 15 characters
serial_number	The format of this number conforms to IBM standard C-S 1-1121-018 1999-06 Serial Numbering for IBM products. The serial number is 7 digits, the first two of which define the manufacturing location, leaving 5 digits for the product. The standard defines a way to extend the serial number using letters in the place of numbers in the 5-digit field.

Data types	Value ranges
ip_address_arg	The decimal, dotted quad notation standard rules. The following Internet Protocol 4 (IPv4) and Internet Protocol 6 (IPv6) address formats are supported: **IPv4 (no port set, SAN Volume Controller uses default)** 1.2.3.4 **IPv4 with specific port** 1.2.3.4:22 **Full IPv6, default port** 1234:1234:0001:0123:1234:1234:1234:1234 **Full IPv6, default port, leading zeros suppressed** 1234:1234:1:123:1234:1234:1234:1234 **Full IPv6 with port** [2002:914:fc12:848:209:6bff:fe8c:4ff6]:23 **Zero-compressed IPv6, default port** 2002::4ff6 **Zero-compressed IPv6 with port** [2002::4ff6]:23
dns_name	Dotted domain name for the subnet that the cluster is in. For example, ibm.com.
hostname	The host name that is assigned to the cluster. This can be different from the cluster name and you can change the host name at any time. A combination of the host name and the dns_name that is used to access the cluster, for example: https://hostname.ibm.com/
capacity_value	A value with a range of 512 bytes up to 2 petabytes (PB). **Note:** The capacity can be specified as MB, KB, GB, or PB. When MB is used, the value is specified in multiples of 512 bytes. A capacity of 0 is valid for a striped or sequential VDisk. The smallest number of supported bytes is 512.
node_id	A node ID differs from other IDs; a node ID is a unique ID that is assigned when a node is used to create a cluster, or when a node is added to a cluster. A *node_id* value is never reused in a cluster. Node IDs are internally represented as 64-bit numbers. Node IDs, like other IDs, cannot be modified by user commands.

Data types	Value ranges
xxx_id	All objects are referred to by unique integer IDs that are assigned by the system when the objects are created. All IDs are represented internally as 32-bit integers. Node IDs are an exception. IDs in the following ranges are used to identify the various types of objects: • node_id: A positive decimal integer greater than or equal to 1 • mdisk_grp_id: 0 - 127 • io_grp_id: 0 - 3 (See Note.) • mdisk_id: 0 - 4095 • vdisk_id: 0 - 8191 • copy_id: 0 - 1 • host_id: 0 - 1023 • flash_const_grp_id: 0 - 255 • remote_const_grp_id: 0 - 255 • fcmap_id: 0 - 4095 • rcrel_id: 0 - 8191 • controller_id: 0 - 63 **Note:** io_group 4 exists but is used only in certain error recovery procedures. These IDs, like node IDs, cannot be modified by user commands. **Note:** IDs are assigned at run time by the system and cannot be relied upon to be the same after, for example, the configuration restoration. Use object names in preference to IDs when you are working with objects.
xxx_list	A colon-delimited list of values of type *xxx*.
wwpn_arg	The fibre-channel worldwide port name (WWPN). This is expressed as a 64-bit hexadecimal number, for example: 1A2B30C67AFFE47B These numbers must consist of the characters 0 - 9, a - f, and A - F. A command fails if you enter WWPN 0 in the command string.
panel_name	A string of up to 6 characters that correspond to the number on the printed label below the display on the front panel of a node in the cluster.
sequence_number	A 32-bit unsigned integer, expressed in decimal.
csi_num_arg	A 32-bit unsigned integer, expressed in decimal.
percentage_arg	An 8-bit unsigned integer, expressed in decimal 0 - 100.
extent_arg	A 32-bit unsigned integer, expressed in decimal.
num_extents_arg	A 32-bit unsigned integer, expressed in decimal.
threads_arg	An 8-bit unsigned integer, expressed in decimal. Valid values are 1, 2, 3, or 4.
velocity_arg	The fabric speed in GBps. Valid values are 1 or 2.
timezone_arg	The ID as detailed in the output of the **svcinfo lstimezones** command.
timeout_arg	The command timeout period. An integer from 0 to 600 (seconds).

Data types	Value ranges
stats_time_arg	The frequency at which statistics are gathered. Valid values are 1 to 60 minutes in increments of 1 minute.
directory_arg	Specifies a directory and or file name filter within the specified directory. The following directory values are valid: • /dumps • /dumps/audit • /dumps/cimom • /dumps/configs • /dumps/elogs • /dumps/feature • /dumps/iostats • /dumps/iotrace • /home/admin/upgrade The file name filter can be any valid filename with or without the wildcard '*'. The file name filter can be appended to the end of one of the previous directory values.
locale_arg	The cluster locale setting. Valid values are 0 and 3. • 0 US English (default) • 3 Japanese
key_arg	A user-defined identifier for an SSH key. A string of up to 30 characters.
user_arg	Specifies the user: either admin or service.
copy_rate	A numeric value of 0 - 100.
copy_type	Specifies the Mirror copy type: Metro or Global.

The maximum number of values that can be entered into a colon-separated list is 128. If more than 128 items are entered into a list an error is returned.

CLI parameters

CLI parameters are found within the syntax diagram.

CLI parameters can be entered in any order except in the following situations:

- The first argument following the command name must be the action that is to be performed.
- Where you are performing an action against a specific object, the object ID or name must be the last argument in the line.

A valid parameter meets the following requirements:

- Parameters can be entered in any order.
- If a parameter has an associated argument, the argument must *always* follow the parameter.
- A parameter *must* start with a '-'; otherwise, it is assumed to be an argument.
- The maximum length of any single parameter that can be entered into the CLI is 128 bytes.

- An argument can contain multiple data items. The maximum number of data items that you can enter into such a list is 128. For a component list, separate the individual items by a colon.

CLI flags

The following flags are common to all command-line interface (CLI) commands.

-? or -h

Print help text. For example, issuing **svcinfo lscluster -h** provides a list of the actions available with the **svcinfo lscluster** command.

-nomsg

When used, this flag prevents the display of the `successfully created` output. For example, if you issue the following command:

```
svctask mkmdiskgrp -ext 16
```

it displays:

MDisk Group, id [6], successfully created

However, if the -nomsg parameter is added, for example:

```
svctask mkmdiskgrp -ext 16 -nomsg
```

the following information is displayed:

```
6
```

This parameter can be entered for any command, but is only acted upon by those commands that generate the `successfully created` outputs. All other commands ignore this parameter.

CLI messages

Ensure that you are familiar with the command-line interface (CLI) messages.

When some commands complete successfully, textual output is normally provided. However, some commands do not provide any output. The phrase `No feedback` is used to indicate that no output is provided. If the command does not complete successfully, an error is generated. For example, if the command has failed as a result of the cluster being unstable, the following output is provided:

- CMMVC5786E The action failed because the cluster is not in a stable state.

Chapter 1. Preparing the SSH client system for the CLI

Before you can issue command-line interface (CLI) commands from the host to the cluster, you must prepare the Secure Shell (SSH) client system.

Microsoft Windows operating systems

The IBM System Storage Productivity Center (SSPC) and the master console for the SAN Volume Controller include the PuTTY client program, which is a Microsoft® Windows® SSH client program. The PuTTY client program can be installed on your SSPC or master console server in one of the following ways:

- If you purchased the SSPC or the master console hardware option from IBM, the PuTTY client program has been preinstalled on the hardware.
- You can use the master console software installation CD to install the PuTTY client program. The SSPC, master console hardware option, and the software-only master console each provide this CD.
- You can use the separate PuTTY client program-installation wizard, **putty-<version>-installer.exe**. You can download the PuTTY client program from the following Web site:

  www.chiark.greenend.org.uk/~sgtatham/putty/

Note: Before you install the PuTTY client program, ensure that your Windows system meets the system requirements. See the *IBM System Storage Productivity Center Introduction and Planning Guide* for system requirements.

If you want to use an SSH client other than the PuTTY client, the following Web site offers SSH client alternatives for Windows:

www.openssh.org/windows.html

IBM AIX operating systems

For IBM AIX 5L™ 5.1 and 5.2 on Power architecture, you can obtain the OpenSSH client from the bonus packs, but you also must obtain its prerequisite, OpenSSL, from the IBM AIX® toolbox for Linux® applications for IBM Power Systems. For AIX 4.3.3, you can obtain the software from the AIX toolbox for Linux applications.

You can also obtain the AIX installation images from IBM developerWorks® at the following Web site:

oss.software.ibm.com/developerworks/projects/openssh

Linux operating systems

The OpenSSH client is installed by default on most Linux distributions. If it is not installed on your system, consult your Linux installation documentation or visit the following Web site:

www.openssh.org/portable.html

The OpenSSH client can run on a variety of additional operating systems. For more information about the openSSH client, visit the following Web site:

www.openssh.org/portable.html

Preparing the SSH client system to issue CLI commands

To issue command-line interface (CLI) commands to the cluster from a host, you must prepare the Secure Shell (SSH) client on the host so that the host is accepted by the SSH server on the cluster.

To use a host that requires a different type of SSH client, OpenSSH for example, follow the instructions for that software.

Perform the following steps to enable your host to issue CLI commands:

1. For the IBM System Storage Productivity Center or master console and Windows hosts:
   a. Generate an SSH key pair using the PuTTY key generator.
   b. Store the SSH clients public key on the cluster (using a browser that points to the SAN Volume Controller Console).
   c. Configure the PuTTY session for the CLI.
2. For other types of hosts:
   a. Follow the instructions that are specific to the SSH client to generate an SSH key pair.
   b. Store the SSH clients public key on the cluster (using a Web browser to point to the SAN Volume Controller Console or the CLI from an already established host).
   c. Follow the instructions that are specific to the SSH client to establish an SSH connection to the SAN Volume Controller cluster.

Chapter 2. Secure Shell

Secure Shell (SSH) is a client-server network application.

Overview

The SAN Volume Controller cluster acts as the SSH server in this relationship. The SSH client provides a secure environment in which to connect to a remote machine. It uses the principles of public and private keys for authentication.

SSH keys are generated by the SSH software. This includes a public key, which is uploaded and maintained by the cluster, and a private key that is kept private to the host that is running the SSH client. These keys authorize specific users to access the administration and service functions on the cluster. Each key is associated with a user-defined ID string that can consist of up to 40 characters. Up to 100 keys can be stored on the cluster. You can also add new IDs and keys or delete unwanted IDs and keys.

Attention: The SAN Volume Controller does not support running multiple SSH sessions concurrently against a single cluster. This can cause the system to lose access to data and cause data to be lost. To avoid running multiple SSH sessions concurrently against a single cluster, do not run scripts that create child processes that run in the background and invoke SAN Volume Controller commands.

SSH is a communication vehicle between the host system and the following components:

- The SAN Volume Controller
- The system on which the SAN Volume Controller Console is installed

Authenticating SSH logins

When you are using AIX hosts, SSH logins are authenticated on the cluster using the RSA-based authentication that is supported in the OpenSSH client that is available for AIX. This scheme is based on public-key cryptography, using an algorithm known commonly as RSA.

Note: The authentication process for non-AIX hosts systems is similar.

With this scheme (as in similar OpenSSH systems on other host types), the encryption and decryption is done using separate keys. This means that it is not possible to derive the decryption key from the encryption key.

Because physical possession of the private key allows access to the cluster, the private key must be kept in a protected place, such as the .ssh directory on the AIX host, with restricted access permissions.

When SSH client (A) attempts to connect to SSH server (B), the key pair authenticates the connection. The key consists of two halves: the public keys and private keys. The SSH client public key is put onto SSH Server (B) using some means outside of the SSH session. When SSH client (A) tries to connect, the private key on SSH client (A) is able to authenticate with its public half on SSH server (B).

Connecting the SAN Volume Controller Console to additional clusters

The master console includes the preinstalled SAN Volume Controller Console Web server and Common Information Model (CIM) Object manager software. This software uses the PuTTY Secure Shell (SSH) client function for the SAN Volume Controller Console to programmatically access the SAN Volume Controller cluster. The master console comes with preinstalled PuTTY SSH keys. You can generate new PuTTY SSH keys that are unique to your master console, copy the private SSH key to the SAN Volume Controller Console directory, and store the public SSH key on all clusters that are connected to the SAN Volume Controller Console.

You can also install the SAN Volume Controller Console on a Windows 2000 server system that you provide. If you intend to install the SAN Volume Controller Console on a host that you supply, you must install PuTTY first, which is a prerequisite for the SAN Volume Controller Console.

Configuring a Secure Shell client system other than PuTTY

If you use a Secure Shell (SSH) client system other than the PuTTY client, you must configure that client system before you can access the command-line interface (CLI).

Perform the tasks that are equivalent to the following steps to configure your non-PuTTY SSH client system:

1. Install the SSH client software on the computer that will host the master console.

   Note: This step is not required for a master console that has the preinstalled PuTTY client.
2. Generate SSH keys on the SSH client system.
3. Configure the PuTTY session, if required, on the SSH client system.
4. If the client system is the master console, copy the private key into the SAN Volume Controller installation directory. If the client system is not the master console, store the private key on the SSH client system.

   Attention: Do not run scripts that create child processes that run in the background and call SAN Volume Controller commands. This can cause the system to lose access to data and to cause data to be lost.
5. Copy the SSH public key to the master console.
6. Store the SSH client public key on the SAN Volume Controller cluster.

You perform step 6 to store the SSH client public key on the SAN Volume Controller when you complete the creation of the SAN Volume Controller cluster. After you define a cluster to the SAN Volume Controller Console and therefore enable SSH communication to the cluster, you can store additional SSH client public keys on the cluster. You can store additional keys through the SAN Volume Controller Console or the CLI.

Generating an SSH key pair using PuTTY

You must generate a Secure Shell (SSH) key pair to use the SAN Volume Controller Console and the command-line interface (CLI).

Perform the following steps to generate SSH keys on the IBM System Storage Productivity Center (SSPC) or master console using the PuTTY key generator (PuTTYgen):

1. Start PuTTYgen by clicking **Start** → **Programs** → **PuTTY** → **PuTTYgen**. The PuTTY Key Generator panel is displayed.
2. Click **SSH-2 RSA** as the type of key to generate.

   Note: Leave the number of bits in a generated key value at 1024.
3. Click **Generate** and then move the cursor around the blank area of the Key section to generate the random characters that create a unique key. When the key has been completely generated, the information about the new key is displayed in the Key section.

   Attention: Do not modify the Key fingerprint or the Key comment fields; this can cause your key to no longer be valid.
4. (Optional) If you are generating SSH keys for a computer other than the SSPC or master console, enter a passphrase in the **Key passphrase** and **Confirm passphrase** fields. The passphrase encrypts the key on the disk; therefore, it is not possible to use the key without first entering the passphrase.

   Attention: If you are generating the key pair for the SSPC or master console, do not enter anything in the Key passphrase or the Confirm passphrase fields.
5. Save the public key by performing the following steps:
   a. Click **Save public key**. You are prompted for the name and location of the public key.
   b. Type `icat.pub` as the name of the public key and specify the location where you want to save the public key. For example, you can create a directory on your computer called *keys* to store both the public and private keys.
   c. Click **Save**.
6. Save the private key by performing the following steps:
   a. Click **Save private key**. The PuTTYgen Warning panel is displayed.
   b. Click **Yes** to save the private key without a passphrase.
   c. Type `icat` as the name of the private key, and specify the location where you want to save the private key. For example, you can create a directory on your computer called *keys* to store both the public and private keys. It is recommended that you save your public and private keys in the same location.
   d. Click **Save**.
7. Close the PuTTY Key Generator window.

Configuring a PuTTY session for the CLI

You must configure a PuTTY session using the Secure Shell (SSH) key pair that you have generated before you can use the command-line interface (CLI).

Attention: Do not run scripts that create child processes that run in the background and call SAN Volume Controller commands. This can cause the system to lose access to data and cause data to be lost.

Perform the following steps to configure a PuTTY session for the CLI:

1. Select **Start** → **Programs** → **PuTTY** → **PuTTY**. The PuTTY Configuration window opens.

2. Click **Session** in the Category navigation tree. The Basic options for your PuTTY session are displayed.
3. Click **SSH** as the Protocol option.
4. Click **Only on clean exit** as the Close window on exit option. This ensures that connection errors are displayed.
5. Click **Connection** → **SSH** in the Category navigation tree. The options controlling SSH connections are displayed.
6. Click **2** as the Preferred SSH protocol version.
7. Click **Connection** → **SSH** → **Auth** in the Category navigation tree. The Options controller SSH authentication are displayed.
8. Click **Browse** or type the fully qualified file name and location of the SSH client and private key in the **Private key file for authentication** field. The file that you specify in this field is the one that you stored in the SAN Volume Controller software (for example, C:\Program Files\IBM\svcconsole\cimom\icat.ppk).
9. Click **Session** in the Category navigation tree. The Basic options for your PuTTY session are displayed.
10. Click **Default Settings** and then click **Save**.
11. Type the name or IP address of the SAN Volume Controller cluster in the **Host Name (or IP Address)** field.
12. Type 22 in the **Port** field. The SAN Volume Controller cluster uses the standard SSH port.
13. Type the name that you want to use to associate with this session in the **Saved Sessions** field. For example, you can name the session SAN Volume Controller Cluster 1.
14. Click **Save**.

You have now configured a PuTTY session for the CLI.

Adding subsequent SSH public keys to the SAN Volume Controller

You can add subsequent Secure Shell (SSH) public keys to the SAN Volume Controller from the SSH Public Key Maintenance panel.

This task assumes that you are at the Welcome panel for the SAN Volume Controller Console.

The SSH key allows the IBM System Storage Productivity Center or the master console (where the SAN Volume Controller Console is running) to access the cluster.

During the cluster creation wizard, you added a SSH key to the cluster. You can add additional SSH keys to grant SSH access to other servers.

Perform the following steps to add additional SSH keys:

1. Click **Clusters** in the portfolio.
2. Click the cluster whose SSH keys you want to maintain.
3. Select **Maintain SSH Keys** from the task list and click **Go**. The SSH Public Key Maintenance panel is displayed.
4. Follow the instructions that are on the SSH Public Key Maintenance panel.
5. Click **Add Key** when you have completed the SSH Public Key Maintenance panel.

After the initial configuration of the cluster has been performed using the SAN Volume Controller Console and at least one SSH client key has been added, the remainder of the configuration can either be performed using the SAN Volume Controller Console or the command-line interface.

Adding SSH keys for hosts other than the IBM System Storage Productivity Center or the master console

You can add Secure Shell (SSH) keys on other hosts.

Perform the following steps to add SSH keys on hosts:

1. Generate the public-private key pair on each host that you want to use the SAN Volume Controller command-line interface. See the information that came with your SSH client for specific details about using the key generation program that comes with your SSH client.
2. Copy the public keys from each of these hosts to the IBM System Storage Productivity Center or the master console.
3. Use the PuTTY secure copy function to copy these public keys from the IBM System Storage Productivity Center or the master console to the cluster.
4. Repeat 3 for each public key that you copied in step 2.

Chapter 3. Copying the SAN Volume Controller software upgrade files using PuTTY scp

PuTTY scp (pscp) provides a file transfer application for secure shell (SSH) to copy files either between two directories on the configuration node or between the configuration node and another host.

To use the pscp application, you must have the appropriate permissions on the source and destination directories on your respective hosts.

The pscp application is available when you install an SSH client on your host system. You can access the pscp application through a command prompt.

Perform the following steps to use the pscp application:

1. Start a PuTTY session.
2. Configure your PuTTY session to access your SAN Volume Controller cluster.
3. Save your PuTTY configuration session. For example, you can name your saved session SVCPUTTY.
4. Open a command prompt.
5. Issue the following command to set the path environment variable to include the PuTTY directory:

   ```
   set path=C:\Program Files\putty;%path%
   ```

   where *Program Files* is the directory where PuTTY is installed.
6. Issue the following command to copy the package onto the node where the CLI runs:

   ```
   directory_software_upgrade_files pscp -load saved_putty_configuration
    software_upgrade_file_name admin@cluster_ip_address:/home/admin/upgrade
   ```

   where *directory_software_upgrade_files* is the directory that contains the software upgrade files, *saved_putty_configuration* is the name of the PuTTY configuration session, *software_upgrade_file_name* is the name of the software upgrade file, and *cluster_ip_address* is the IP address of your cluster.

   If there is insufficient space to store the software upgrade file on the cluster, the copy process fails. Perform one of the following steps to provide sufficient space:

   - Issue the **svctask cleardumps** CLI command to free space on the cluster and repeat step 6.
   - Issue the following command from the cluster to transfer the error logs to the IBM System Storage Productivity Center or the master console:

   ```
   pscp -unsafe -load saved_putty_configuration
    admin@cluster_ip_address:/dump/elogs/* your_preferred_directory
   ```

   where *saved_putty_configuration* is the name of the PuTTY configuration session, *cluster_ip_address* is the IP address of your cluster, and *your_preferred_directory* is the directory where you want to transfer the error logs.

   After you have transferred the error logs to the IBM System Storage Productivity Center or the master console, repeat step 6.

Chapter 4. Audit log commands

An audit log keeps track of action commands that are issued through a Secure Shell (SSH) session or through the SAN Volume Controller Console.

The audit log entries provide the following information:

- Identity of the user who issued the action command
  - From the command-line interface, the user name (administrator or service), and the label that is associated with the user's public SSH key in the authorized keys file
  - From the native Web pages, the user's identity (admin[web] or service[web]) according to which user name the user authenticated with
  - From the SAN Volume Controller Console, the user's identity (administrator), the label that is associated with the CIMOM key in the authorized keys file, and the user name that has been recorded by the CIMOM when the SAN Volume Controller Console user authenticated with the CIMOM
- The name of the actionable command
- The timestamp of when the actionable command was issued on the configuration node
- The parameters which were issued with the actionable command

The following commands are not documented in the audit log:

- svctask dumpconfig
- svctask cpdumps
- svctask cleardumps
- svctask finderr
- svctask dumperrlog
- svctask dumpinternallog
- svcservicetask dumperrlog
- svcservicetask finderr

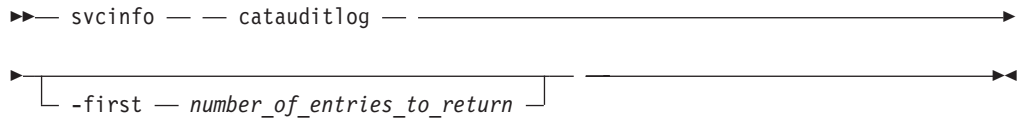
The following items are also not documented in the audit log:

- Commands that fail are not logged
- A result code of 0 (success) or 1 (success in progress) is not logged
- Result object ID of node type (for the addnode command) is not logged
- Views are not logged

catauditlog

Use the **catauditlog** command to display the in-memory contents of the audit log.

Syntax



Parameters

-first *number_of_entries_to_return*
(Optional) Specifies the number of most recent entries to display.

Description

This command lists a specified number of the most recently audited commands.

The in-memory portion of the audit log can hold approximately 1 MB of audit information. Depending on the command text size and the number of parameters, 1 MB records approximately 6000 commands.

Once the in-memory audit log has reached its maximum capacity, the log is written to a local file on the configuration node in the /dumps/audit directory. The **svcinfo catauditlog** command only displays the in-memory part of the audit log; the on-disk part of the audit log is in readable text format and does not need any special command to decode it.

The in-memory log entries are reset and cleared automatically, ready to start accumulating new commands. The on-disk portion of the audit log can then be analyzed at a later date.

The lsauditlogdumps command can be used to list the files that are on the disk.

The in-memory portion of the audit log can be transferred to an on-disk file using the **svctask dumpauditlog** command. This action clears the in-memory portion of the log.

In the following example, the user has specified that they want to list the 15 most recent audit log entries.

An invocation example

```
svcinfo catauditlog -delim : -first 15
```

The resulting output

```
audit_seq_no:timestamp:cluster_user:ssh_label:icat_user:result:res_obj_id
:action_cmd
125:060311111800:admin:Joe::0::svctask rmsshkey -key label47 -user admin
126:060311111800:admin:Joe::0::svctask addsshkey -label label48 -file
/home/Joe/id_rsa.pub -user admin
127:060311111800:admin:Joe::0::svctask rmsshkey -key label48 -user admin
128:060311111800:admin:Joe::0::svctask addsshkey -label label49 -file
/home/Joe/id_rsa.pub -user admin
129:060311111800:admin:Joe::0::svctask rmsshkey -key label49 -user admin
130:060311134617:admin:Joe::0::svctask chmdisk -name ca-0 1
131:060311134617:admin:Joe::0::svctask chmdisk -name ca-1 2
132:060311134617:admin:Joe::0::svctask chmdisk -name ca-2 3
133:060311134617:admin:Joe::0::svctask chmdisk -name cb-0 4
134:060311134617:admin:Joe::0::svctask chmdisk -name cb-1 5
135:060311134617:admin:Joe::0::svctask chmdisk -name cb-2 6
136:060311134617:admin:Joe::0::svctask chmdisk -name cc-0 7
137:060311134617:admin:Joe::0::svctask chmdisk -name cc-1 8
138:060311134617:admin:Joe::0::svctask chmdisk -name cc-2 9
139:060311134632:admin:Joe::0::svctask mkmdiskgrp -name custa-mdisks -ext 512
-mdisk ca-0:ca-1:ca-2
```

dumpauditlog

Use the **dumpauditlog** command to reset or clear the contents of the in-memory audit log. The contents of the audit log are sent to a file in the /dumps/audit directory on the current configuration node.

Syntax

Parameters

There are no parameters.

Description

This command dumps the contents of the audit log to a file on the current configuration node. It also clears the contents of the audit log. This command is logged as the first entry in the new audit log.

Audit log dumps are automatically maintained in the /dumps/audit directory. The local file system space is used by audit log dumps and is limited to 200 MB on any node in the cluster. The space limit is maintained automatically by deleting the minimum number of old audit log dump files so that the /dumps/audit directory space is reduced below 200 MB. This deletion occurs once per day on every node in the cluster. The oldest audit log dump files are considered to be the ones with the lowest audit log sequence number. Also, audit log dump files with a cluster ID number that does not match the current one are considered to be older than files that match the cluster ID, regardless of sequence number.

Other than by running dumps (or copying dump files among nodes), you cannot alter the contents of the audit directory. Each dump file name is generated automatically in the following format:

`auditlog_`*`firstseq_lastseq_timestamp_clusterid`*

where

- *firstseq* is the audit log sequence number of the first entry in the log
- *lastseq* is the audit sequence number of the last entry in the log
- *timestamp* is the timestamp of the last entry in the audit log that is being dumped
- *clusterid* is the cluster ID at the time that the dump was created

The audit log dump files names cannot be changed.

The audit log entries in the dump files contain the same information as displayed by the **svcinfo catauditlog** command; however, the **svctask dumpauditlog** command displays the information with one field per line. The **svcinfo lsauditlogdumps** command displays a list of the audit log dumps that are available on the nodes in the cluster.

An invocation example

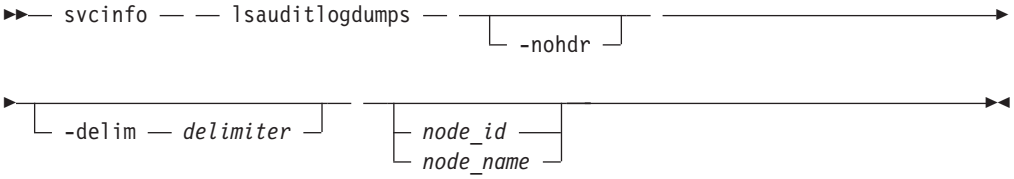
```
svctask dumpauditlog
```

The resulting output

lsauditlogdumps

The **lsauditlogdumps** command generates a list of the audit log dumps that are available on the nodes in the cluster.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a 1-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

node_id | node_name
(Optional) Specifies the node ID or name to list the available dumps of the given type. If you do not specify a node, the files on the current configuration node are displayed.

Description

This command lists the dump files that are in the **/dumps/audit** directory on the specified node, or on the configuration node if a node is not specified.

The cluster automatically creates the audit log. The audit log can also be created manually by issuing the **svctask dumpauditlog** command. The audit log comprises the files that are listed by the **svcinfo lsauditlogdumps** command. These files are limited to approximately 200 MB on each node in the cluster, at which point the oldest files are automatically deleted. When the configuration node changes to a different node in the cluster, any old audit log files are left on the former configuration node. As with other types of dumps, you can retrieve those files using the **cpdumps** command.

An invocation example

```
svcinfo lsauditlogdumps
```

The resulting output

```
id auditlog_filename
0 auditlog_0_229_060311234532_0000020060013d8a
1 auditlog_230_475_060312234529_0000020060013d8a
2 auditlog_476_491_060313234527_0000020060013d8a
```

Audit log dump file contents

```
...
Auditlog Entry:23
Audit Sequence Number :138
Timestamp :Sat Mar 11 13:46:17 2006
:Epoch + 1142084777
SVC User :admin
SSH Label :Joe
ICAT User :
Result Object ID :
Result Code :0
Action Command :svctask chmdisk -name cc-2 9
Auditlog Entry:24
Audit Sequence Number :139
Timestamp :Sat Mar 11 13:46:32 2006
:Epoch + 1142084792
SVC User :admin
SSH Label :Joe
ICAT User :
Result Object ID :
Result Code :0
Action Command :svctask mkmdiskgrp -name custa-mdisks -ext
512 -mdisk ca-0:ca-1:ca-2
...
```

Chapter 5. Role-based security commands

Role-based security commands restrict administrative abilities by assigning one of three roles: **Administrator** (admin), **CopyOperator**, or **Monitor** (the default).

Use the following three commands to manage role assignments:

mkauth
Assigns the role of **CopyOperator** or **Administrator**

rmauth
Reverts the assigned role to the default role of **Monitor**

lsauth Displays entries in the Role-based Security authorization table

The **Administrator** and **CopyOperator** roles apply only to SSH sessions that are established within the SAN Volume Controller cluster by an **Administrator**. The commands that a user can initiate are determined by the role that is associated with the SSH key that established the session.

As an **Administrator**, a user can initiate all commands, and perform configurations and backups using the **svcconfig** backup and restore tool.

By default, **Administrator** users also have the role of **Monitor**. As a **Monitor**, a user can initiate the following SAN Volume Controller CLI commands and functions:

svcinfo commands
All **svcinfo** commands

svctask commands
Only the following commands: **finderr**, **dumperrlog**, and **dumpinternallog**

svcservicetask commands
Only the following commands: **finderr** and **dumperrlog**

svcconfig backup and restore tool
A configuration backup can be performed; a configuration restore can only be performed by an **Administrator**.

An **Administrator** can also have the role of **CopyOperator**. As a **CopyOperator**, a user can initiate the following SAN Volume Controller CLI commands and functions:

svcinfo commands
All **svcinfo** commands can be initiated.

svctask commands
Only the following commands can be initiated: **finderr**, **dumperrlog**, **dumpinternallog**, **prestartfcconsistgrp**, **startfcconsistgrp**, **stopfcconsistgrp**, **chfcconsistgrp**, **prestartfcmap**, **startfcmap**, **stopfcmap**, **chfcmap**, **startrcconsistgrp**, **stoprcconsistgrp**, **switchrcconsistgrp**, **chrcconsistgrp**, **startrcrelationship**, **stoprcrelationship**, **switchrcrelationship**, **chrcrelationship**, **chpartnership**

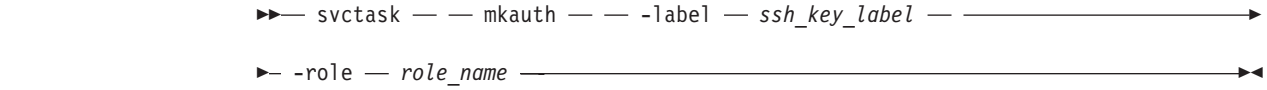
svcservicetask commands
Only the following commands can be initiated: **finderr** and **dumperrlog**

svcconfig backup and restore tool
A configuration backup can be performed; a configuration restore can only be performed by an **Administrator**.

mkauth

The **mkauth** command changes a user's authorization role from **Monitor** to **CopyOperator** or **Administrator**.

Syntax



Parameters

-label *ssh_key_label*
(Required) Specifies the identifier that is associated with the secure shell (SSH) key of the user whose authorization is being created. The SSH key label is the same identifier that you supplied with the **-label** parameter in the **svctask addsshkey** command.

-role *role_name*
(Required) Specifies the name of the role that is assigned to the user, either **CopyOperator** or **Administrator**.

Description

The **mkauth** command allows you to change the default role of **Monitor** to either **CopyOperator** or **Administrator**. The roles that are assigned by the **mkauth** command apply only to SSH sessions that have been established within the SAN Volume Controller cluster by an **Administrator**. The commands that you can initiate in an assigned role are determined by the role that is associated with the SSH key that established the session.

Note: Only one non-default role is supported for each SSH key. If the specified SSH key already has a non-default role, the **mkauth** command fails.

The **CopyOperator** role allows a user to initiate the following SAN Volume Controller CLI commands and functions:

svcinfo commands
All **svcinfo** commands

svctask commands
Only the following commands: **finderr**, **dumperrlog**, **dumpinternallog**, **prestartfcconsistgrp**, **startfcconsistgrp**, **stopfcconsistgrp**, **chfcconsistgrp**, **prestartfcmap**, **startfcmap**, **stopfcmap**, **chfcmap**, **startrcconsistgrp**, **stoprcconsistgrp**, **switchrcconsistgrp**, **chrcconsistgrp**, **startrcrelationship**, **stoprcrelationship**, **switchrcrelationship**, **chrcrelationship**, **chpartnership**

svcservicetask commands
Only the following commands: **finderr** and **dumperrlog**

svcconfig backup and restore tool
A configuration backup can be performed; a configuration restore can only be performed by an **Administrator**.

As an **Administrator**, a user can initiate all commands, and perform configurations and backups using the **svcconfig** backup and restore tool.

An invocation example

In the following example, the user who uses an SSH key with the label "rob" is assigned the role of **CopyOperator**.

```
svctask mkauth -label rob -role CopyOperator
```

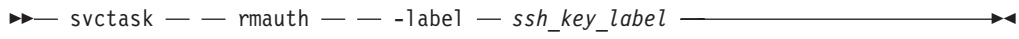
The resulting output

```
[No feedback]
```

rmauth

The **rmauth** command removes the authorization that is assigned to **CopyOperator** and **Administrator** users.

Syntax



Parameters

-label *ssh_key_label*
(Required) Specifies the identifier that is associated with the secure shell (SSH) key of the user whose role authorization is being removed. The SSH key label is the same identifier that is supplied with the **-label** parameter in the **svctask addsshkey** command.

Description

The **rmauth** command removes the explicit authorization that is associated with the SSH key when the role of **CopyOperator** or **Administrator** is assigned. The specified SSH key reverts to the default **Monitor** authorization.

The **Monitor** default role allows a user to initiate the following SAN Volume Controller CLI commands and functions:

svcinfo commands
All **svcinfo** commands.

svctask commands
Only the following commands: **finderr**, **dumperrlog**, and **dumpinternallog**

svcservicetask commands
Only the following commands: **finderr** and **dumperrlog**

svcconfig backup and restore tool
A configuration backup can be performed; a configuration restore can only be performed by an **Administrator**.

An invocation example

In the following example, the authorization user who uses an SSH key with the label "rob" is removed. The authorization level for this user reverts to Monitor.

```
svctask rmauth -label rob
```

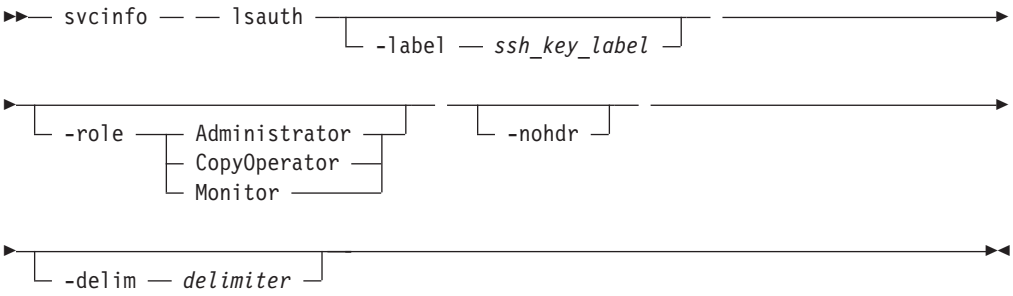
The resulting output

```
[No feedback]
```

lsauth

The **lsauth** command generates a report that lists the default and non-default authorizations for the **admin** user SSH keys. A list of these authorizations also reside in the Role Based Security authorization table.

Syntax



Parameters

-label *ssh_key_label*
(Optional) Specifies the identifier that is associated with the secure shell (SSH) key. The SSH key label is the same identifier that you supplied with the **-label** parameter in the **svctask addsshkey** command.

Note: You cannot use the **-label** and **-role** parameters together.

-role Administrator | CopyOperator | Monitor
(Optional) Generates a report that displays a specific role. You can only specify one role at a time. You must use the capitalization that is specified in the syntax diagram for the designated roles.

You cannot use the **-role** and **-label** parameters together.

-nohdr
(Optional) By default, headings are displayed for each column of data (in a concise style view), and for each item of data (in a detailed style view). Using the **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are separated with spaces. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you entered `-delim :`, a colon character (:) separates all items of data in a concise view (the spacing of columns does not occur). In a detailed view, the data is separated from its header by the colon character (:).

Description

The **lsauth** command generates a report that displays the entries that are found in the Role Based Security authorization table. If either the **-label** or the **-role** parameter is specified, the output consists only of entries that match the specified criteria.

An invocation example

```
svcinfo lsauth
```

The resulting output

```
id  ssh_label     Role
0    key0     Administrator
1    key1     CopyOperator
2    key2     Monitor
```

Chapter 6. E-mail and inventory event notification commands

You can use the command-line interface (CLI) to enable your system to send notifications.

You must supply the IP address of the server that receives the notifications, and you must ensure that the server is running. The SAN Volume Controller cluster does perform this type of a check. You can have up to twelve recipients of the notifications.

To enable automatic reporting of errors and inventory information to IBM using call home, create an e-mail user with the following settings.

Note: Inventory information is automatically reported to IBM when you activate error reporting.

usertype
support

errtype
critical_only

enableinventory
yes

email address

- If the SAN Volume Controller is located in North America, Latin America, South America or the Caribbean Islands, callhome1@de.ibm.com
- If the SAN Volume Controller is located anywhere else in the world, callhome0@de.ibm.com

The following e-mail and inventory event notification commands are available:

chcluster
Enables the inventory notification function by specifying how often notifications are sent to recipients. See the "chcluster" on page 38 command in the Chapter 7, "Cluster commands," on page 35 section.

chemail
Modifies settings that are used by the e-mail notification function.

chemailuser
Allows you to modify the e-mail recipient's information.

lscluster
Provides information about the addition of e-mail notification recipients to the cluster. See the "lscluster" on page 197 command in the Chapter 20, "Information commands," on page 191 section.

lsemailuser
Generates a report that lists the users (by user ID or type of user [local or support]) who are receiving e-mail and inventory notifications.

mkemailuser
Allows you to specify the user, the receiving server address, the type of e-mail notification, and to enable inventory notification in the e-mail notification function.

rmemailuser
Removes an existing e-mail recipient from the e-mail notification function.

sendinventoryemail
Sends an inventory e-mail notification to all inventory e-mail recipients.

setemail
Configures your system to use the e-mail notification function.

startemail
Starts the e-mail notification function.

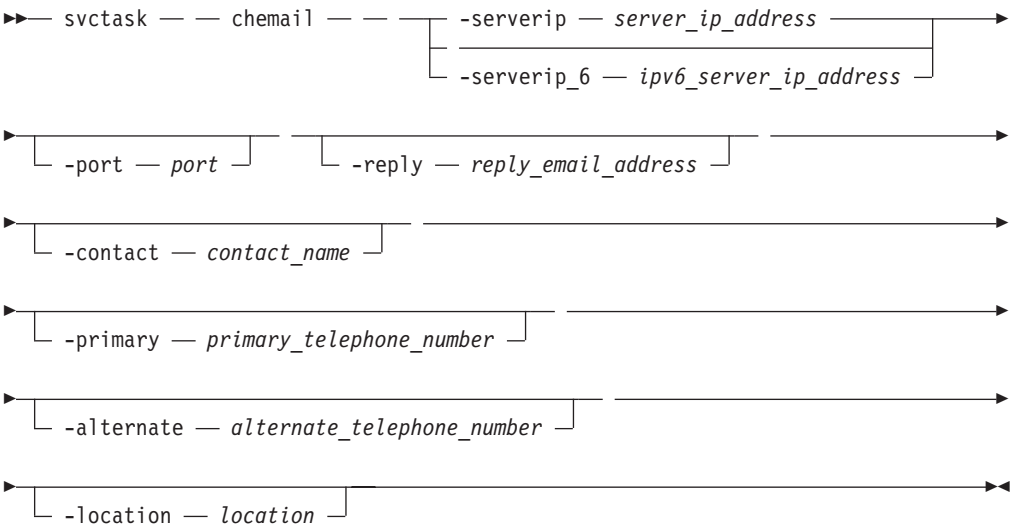
stopemail
Disables the e-mail notification function.

testemail
Sends a test e-mail to a specified user or to all users using the e-mail notification function.

chemail

The **chemail** command allows you to modify the settings that are used by the e-mail notification function. To process this command, at least one of the optional parameters must be specified.

Syntax



Parameters

-serverip *server_ip_address*
(Optional) Specifies the IPv4 address of the SMTP e-mail server that is used for e-mail notification.

-serverip_6 *ipv6_server_ip_address*
(Optional) Specifies the IPv6 address of the SMTP e-mail server that is used for e-mail notification.

-port *port*
(Optional) Specifies the port number that is used for the server IP address. The default port value for SMTP is 25.

-reply *reply_email_address*
(Optional) Specifies the e-mail address to which a reply is sent.

-contact *contact_name*
(Optional) Specifies the name of the person to receive the e-mail.

-primary *primary_telephone_number*
(Optional) Specifies the primary contact telephone number.

-alternate *alternate_telephone_number*
(Optional) Specifies the alternate contact telephone number that is used when you cannot reach the primary contact on the primary phone.

-location *location*
(Optional) Specifies the physical location of the system that is reporting the error. The *location* value must not contain punctuation or any other characters that are not alphanumeric or spaces.

Description

This command modifies the specified settings that are used by the e-mail notification function. This command can be run only if the **setemail** command has been run previously. Issue the **lscluster** command to see the state of the e-mail notification function. If a state of stopped or running is displayed, you can successfully process the **chemail** command. If a state of invalid is displayed, you cannot process the **chemail** command, but you can use the **setemail** command instead.

An invocation example

```
svctask chemail -serverip 9.20.153.255 -port 25 -primary 0441234567
-contact 'manager2008' -reply manager2008@ibm.com
-location 'room 256 floor 1 IBM'
```

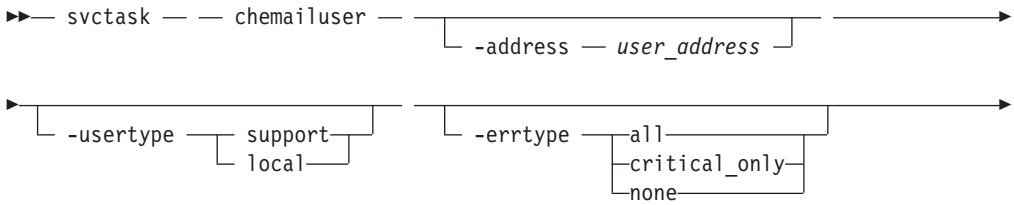
The resulting output

```
[No feedback]
```

chemailuser

The **chemailuser** command modifies the settings that are defined for an e-mail recipient.

Syntax



Parameters

-address *user_address*
(Optional) Specifies the e-mail address of the person receiving the e-mail and or inventory notifications.

-usertype support | local
(Optional) Specifies the type of user, either local or support, based on the following definitions:

support
Address of the support organization that provides vendor support. You cannot specify **support** if the **-errtype** parameter is specified as **all**.

local All other addresses.

-errtype all | critical_only | none
(Optional) Specifies the types of error for which a recipient receives e-mail notification. Recipients can receive the following types of e-mail notifications:

all The recipient receives an e-mail for all errors and events that are logged. You cannot specify this error type if you have specified that the user type is **support**.

critical_only
The recipient receives an e-mail only for critical errors. Warning and informational errors are not included in the e-mail.

none No e-mails are sent to this recipient.

-user *user_name*
(Optional) Specifies the user name of the new e-mail notification recipient. The *user_name* value must be unique, must not contain spaces, and must not contain all numbers. The name **emailuser*n***, where *n* is a number, is reserved and cannot be specified as one of your user names.

-inventory on | off
(Optional) Specifies whether this recipient receives inventory e-mail notifications.

userid_or_name
(Required) Specifies the e-mail recipient for whom you are modifying settings.

Description

This command modifies the settings that are established for an e-mail recipient. Standard rules regarding names apply; therefore, it is not possible to change a name to **emailuser***n*, where *n* is a number. Support users must have the error type set to **Critical_only**. If the wrong error type is assigned to a user type, the command fails.

An invocation example

The following example modifies e-mail settings for e-mail recipient **manager2008**:

```
svctask chemailuser -usertype local manager2008
```

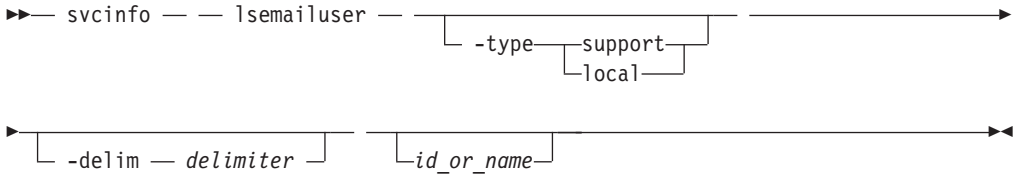
The resulting output

```
[no feedback]
```

lsemailuser

The **lsemailuser** command generates a report that lists the e-mail notification settings for all e-mail recipients, an individual e-mail recipient, or a specified type (local or support) of e-mail recipient.

Syntax



Parameters

-type support | local
(Optional) Specifies the types of e-mail recipients you want to view, either customer or support based as determined by the following definitions:

support
Address of the support organization that provides vendor support. You cannot specify **support** if the **-errtype** parameter is specified as **all**.

local All other addresses.

You cannot use this parameter with the *id_or_name* parameter.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, a colon separates all items of data in a concise view; the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon.

id_or_name
(Optional) Specifies the user ID or user name of the e-mail recipient for whom you want to see the e-mail notification settings. You cannot use this parameter with the **-type** parameter.

Description

When you issue this command, a report is displayed that lists the e-mail notification settings for all e-mail recipients, an individual e-mail recipient, or a specified type (local or support) of e-mail recipient. The concise and detailed views report the same information.

An invocation example

The following command lists information for all e-mail recipients using the e-mail notification function, in a concise view:

```
svcinfo lsemailuser -delim :
```

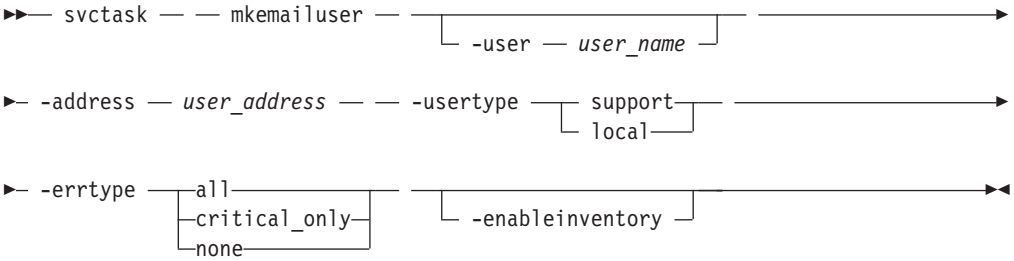
The resulting output

```
id:name:address:err_type:user_type:inventory
1:Support:callhome1@de.ibm.com:critical_only:support:on
2:Fred:fred_house@my_company.co.uk:critical_only:local:on
3:Log:our_log@my_company.co.uk:all:local:off
4:Anne:anne@my_company.co.uk:none:local:off
```

mkemailuser

The **mkemailuser** command adds a recipient of e-mail and inventory notifications to the e-mail notification function. You can add up to twelve recipients, one recipient at a time.

Syntax



Parameters

-user *user_name*
(Optional) Specifies the name of the person that is the recipient of e-mail notifications. The *user_name* value must be unique, must not contain spaces, and must not contain only numbers. If you do not specify a user name, the system automatically assigns a user name in the format of **emailuser***n*, where *n* is a number beginning with 0 (**emailuser0**, **emailuser1**, and so on).

The name **emailuser***n*, where *n* is a number, is reserved and cannot be used as one of your user names.

-address *user_address*
(Required) Specifies the e-mail address of the person receiving the e-mail or inventory notifications, or both.

-usertype support | local
(Required) Specifies the type of user, either customer or support, based on the following definitions:

support
Address of the support organization that provides vendor support. You cannot specify **support** if the **-errtype** parameter is specified as **all**.

local All other addresses.

-errtype all | critical_only | none
(Required) Specifies the types of error for which a recipient receives e-mail notification. Recipients can receive the following types of e-mail notifications:

all The recipient receives an e-mail for all errors and events that are logged. You cannot specify this error type if you have specified that the user type is **support**.

critical_only
The recipient receives an e-mail only for critical errors. (Warning and informational errors are not included in the e-mail.)

none No e-mails are sent to this recipient.

-enableinventory
(Optional) Specifies that this recipient is to receive inventory e-mail notifications.

Description

This command adds e-mail recipients to the e-mail and inventory notification function. You can add up to twelve recipients, one recipient at a time. When an e-mail user is added, if a user name is not specified, a default name is allocated by the system. This default name has the form of **emailuser1**, **emailuser2**, and so on. E-mail notification starts when you process the **startemail** command.

An invocation example

In the following example, you are adding e-mail recipient **manager2008** and designating that **manager2008** receive notification e-mails that contain all error types:

```
svctask mkemailuser -address manager2008@ibm.com -errtype all -usertype local
```

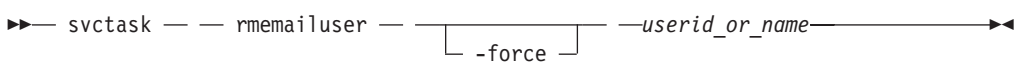
The resulting output

```
email user, id [2], successfully created
```

rmemailuser

The **rmemailuser** command allows you to remove a previously defined e-mail recipient from your system.

Syntax



Parameters

-force
(Optional) Specifies that you want to remove the designated user even though the user is the last e-mail recipient in the system. If you try to remove the last recipient using the e-mail notification function and do not use the **-force** parameter, the transaction fails.

userid_or_name
(Required) Specifies the user ID or user name of the e-mail recipient to remove.

Description

This command removes an existing e-mail recipient from the system. You must use the **-force** parameter to remove the last e-mail recipient from the e-mail notification function.

An invocation example

The following example removes e-mail recipient **manager2008**:

```
svctask rmemailuser manager2008
```

The resulting output

```
[No feedback]
```

An invocation example

The following example removes e-mail recipient **2**:

```
svctask rmemailuser 2
```

The resulting output

```
[No feedback]
```

sendinventoryemail

The **sendinventoryemail** command sends an inventory e-mail notification to all e-mail recipients that are enabled to receive inventory e-mail notifications. There are no parameters for this command.

Syntax

Parameters

There are no parameters for this command.

Description

This command sends an inventory e-mail notification to all e-mail recipients that are enabled to receive inventory e-mail notifications. This command fails if the **startemail** command has not been processed and at least one e-mail recipient using the e-mail and inventory notification function has not been set up to receive inventory e-mail notifications. This command also fails if the email infrastructure has not been set up using the **setemail** command.

An invocation example

In the following example, you send an inventory e-mail notification to all e-mail recipients that are enabled to receive them:

```
svctask sendinventoryemail
```

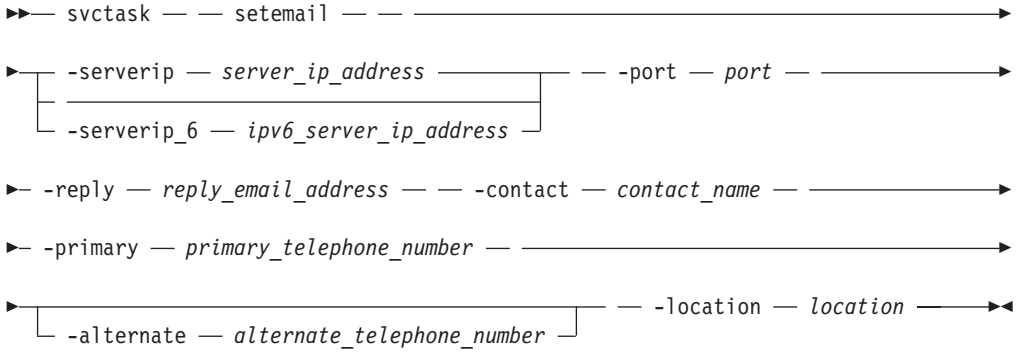
The resulting output

```
[No feedback]
```

setemail

The **setemail** command allows you to configure your system to use the e-mail notification function. It also allows you to specify which data to include in the header section of all e-mail messages and the details of the server that sends the e-mail.

Syntax



Parameters

-serverip *server_ip_address*
(Optional) Specifies the IPv4 address of the SMTP e-mail server that is used for e-mail notification.

-serverip_6 *ipv6_server_ip_address*
(Optional) Specifies the IPv6 address of the SMTP e-mail server that is used for e-mail notification.

-port *port*
(Required) Specifies the port number that is used for the server IP address. The default port value for SMTP is 25.

-reply *reply_email_address*
(Required) Specifies the e-mail address to which a reply is sent.

-contact *contact_name*
(Required) Specifies the name of the person to be contacted.

-primary *primary_telephone_number*
(Required) Specifies the primary contact telephone number.

-alternate *alternate_telephone_number*
(Optional) Specifies the alternate contact telephone number that is used when you cannot reach your primary contact on the primary phone.

-location *location*
(Required) Specifies the physical location of the system that is reporting the error. The *location* value must not contain punctuation or any other characters that are not alphanumeric or spaces.

Description

This command initializes the settings that are used by the e-mail notification function. These settings define the SMTP server that is used to initiate the e-mail notification and the header information that is included in all e-mail. You must use

this command to configure the system before you can start the e-mail notification function.

An invocation example

```
svctask setemail -serverip 9.20.153.255 -port 25 -primary 01234567890
-contact 'manager2008' -reply manager2008@ibm.com
-location 'room 256 floor 1 IBM'
```

The resulting output

```
[No feedback]
```

startemail

The **startemail** command activates the e-mail and inventory notification function. There are no parameters for this command.

Syntax

Parameters

There are no parameters for this command.

Description

This command enables the e-mail error notification service. No e-mails are sent to users until the **startemail** command has been run and at least one user has been defined to the system.

An invocation example

In the following example, you are starting the e-mail error notification service.

```
svctask startemail
```

The resulting output

```
[No feedback]
```

stopemail

The **stopemail** command stops the e-mail and inventory notification function. There are no parameters for this command.

Syntax

Parameters

There are no parameters for this command.

Description

This command stops the e-mail error notification function. No e-mails are sent to users until the **startemail** command is reissued.

An invocation example

In the following example, you have stopped the e-mail and inventory notification function:

```
svctask stopemail
```

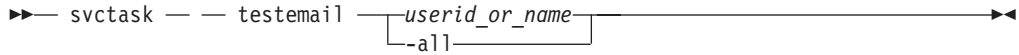
The resulting output

```
[No feedback]
```

testemail

The **testemail** command allows you to send an e-mail notification to one user of the e-mail notification function or to all users of the e-mail notification function to ensure that the function is operating correctly.

Syntax



Parameters

userid_or_name
(Required if you do not specify **-all**) Specifies the user ID or user name of the e-mail recipient that you want to send a test e-mail to. You cannot use this parameter with the **-all** parameter. The *userid_or_name* value must not contain spaces.

-all
(Required if you do not specify *userid_or_name*) Sends a test e-mail to all the e-mail recipients who are not assigned an error type of **none**, and who are assigned the use of the e-mail notification function. You cannot use this parameter with the *userid_or_name* parameter.

Description

This command sends test e-mails to the specified e-mail users. The e-mail recipient expects to receive the test e-mail within a specified service time. If the e-mail is not received within the expected time period, the recipient must contact the administrator to ensure that the e-mail settings for the user are correct. If there is still a problem, you must contact the IBM Support Center.

The e-mail recipient uses the test e-mail to check that the SMTP name, the IP address, the SMTP port, and the user address are valid.

An invocation example

The following example sends a test e-mail to the user ID **manager2008**:

```
svctask testemail manager2008
```

The resulting output

```
[No feedback]
```

Chapter 7. Cluster commands

Cluster commands are used to monitor and modify clusters.

A cluster consists of up to four pairs of nodes that provide a single configuration and service interface. There are a number of cluster commands available for various tasks.

addnode
Adds a new (candidate) node to an existing cluster.

chcluster
Modifies the attributes of an existing cluster.

chiogrp
Modifies the name of an I/O group or the amount of memory that is available for Copy Services or VDisk mirroring.

chnode
Changes the name assigned to a node.

cleardumps
Cleans the various dump directories on a specified node.

cpdumps
Copies dump files from a nonconfiguration node onto the configuration node.

detectmdisk
Rescans the fibre-channel network for new managed disks.

rmnode
Deletes a node from the cluster.

setclustertime
Sets the time for the cluster.

setpwdreset
Changes the status of the password-reset feature for the display panel.

settimezone
Sets the time zone for the cluster.

startstats
Starts the collection of statistics for VDisks and MDisks.

stopcluster
Shuts down a single node or the entire cluster.

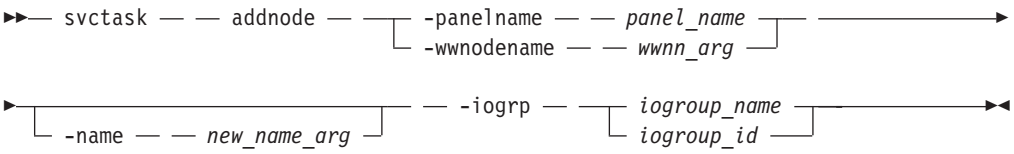
stopstats
Stops the collection of statistics for VDisks and MDisks.

addnode

You can use the **addnode** command to add a new (candidate) node to an existing cluster. You can enter this command any time after a cluster has been created. If you are adding a new node to a cluster, you must ensure that the model type of the new node is supported by the SAN Volume Controller software version of the

cluster. If the model type is not supported by the cluster software, you must upgrade the cluster to a software version that supports the model type of the new node.

Syntax



Parameters

-panelname *panel_name*
(Required if you do not specify the **-wwnodename** parameter) Specifies the node that you want to add to a cluster by the name that is displayed on the display panel. You cannot use this parameter with the **-wwnodename** parameter.

-wwnodename *wwnn_arg*
(Required if you do not specify the **-panelname** parameter) Specifies the node that you want to add to the cluster by the worldwide node name (WWNN). You cannot use this parameter with the **-panelname** parameter.

-name *new_name_arg*
(Optional) Specifies a name for the node that you want to add to the cluster.

-iogrp *iogroup_name* | *iogroup_id*
(Required) Specifies the I/O group to which you want to add this node.

Description

This command adds a new node to the cluster. You can obtain a list of candidate nodes (those that are not already assigned to a cluster) by typing `svcinfo lsnodecandidate`.

Before you add a node to the cluster, you must check to see if any of the following conditions are true. If the following conditions exist, failure to follow the procedures that are documented here might result in the corruption of all data that is managed by the cluster.

- Does the node being added to the cluster use physical node hardware or a slot that has previously been used for a node in the cluster?
- Does the node being added to the cluster use physical node hardware that has been used as a node in another cluster, and are both clusters recognized by the same hosts?

If any of the previous conditions are true, you must take the following actions:

1. Add the node to the same I/O group that it was previously in. You can use the command-line interface command **svcinfo lsnode** or the SAN Volume Controller Console to determine the WWNN of the cluster nodes.
2. Shut down all of the hosts that use the cluster, before you add the node back into the cluster.
3. Add the node back to the cluster before the hosts are restarted. If the I/O group information is unavailable or it is inconvenient to shut down and restart all of the hosts that use the cluster, you can do the following:

a. On all of the hosts that are connected to the cluster, unconfigure the fibre-channel adapter device driver, the disk device driver, and the multipathing driver before you add the node to the cluster.
b. Add the node to the cluster and then reconfigure the fibre-channel adapter device driver, the disk device driver, and multipathing driver.

If you are adding a new node to a cluster, take the following actions:

1. Ensure that the model type of the new node is supported by the SAN Volume Controller software version of the cluster. If the model type is not supported by the cluster software, you must upgrade the cluster to a software version that supports the model type of the new node.
2. Record the node serial number, the WWNN, all WWPNs, and the I/O group to which the node has been added. You might need to use this information later. Having it available can prevent possible data corruption if the node must be removed from and readded to the cluster.

Node addition completes asynchronously; while the node is in the adding state, the WWPN is not known and displays as zeros.

If the compatibility check fails, the following message displays:

CMMVC6201E The node could not be added, because incompatible software: status code [%1].

Other considerations when you add a node to a cluster:

When you add a node to the cluster using the **svctask addnode** command or the cluster GUI, you must confirm whether the node has previously been a member of the cluster. If it has, follow one of these two procedures:

- Add the node to the same I/O group that it was previously in. You can determine the WWNN of the nodes in the cluster using the **svcinfo lsnode** command.
- If you cannot determine the WWNN of the nodes in the cluster, call the support team to add the node back into the cluster without corrupting the data.

When a node is added to a cluster, it displays a state of adding. It can take as long as 30 minutes for the node to be added to the cluster, particularly if the software version of the node has changed.

Attention: If the node remains in the adding state for more than 30 minutes, contact your support representative to assist you in resolving this issue.

Optionally, you can assign a name to the new node. You can use this name in subsequent commands to refer to the node, instead of using the node ID. If you assign a label, this label is displayed as the node name from then on. If you do not assign a label, the default label is `nodeX`, where *X* is the node ID.

An invocation example

```
svctask addnode -wwnodename 5005076801e08b -iogrp io_grp0
```

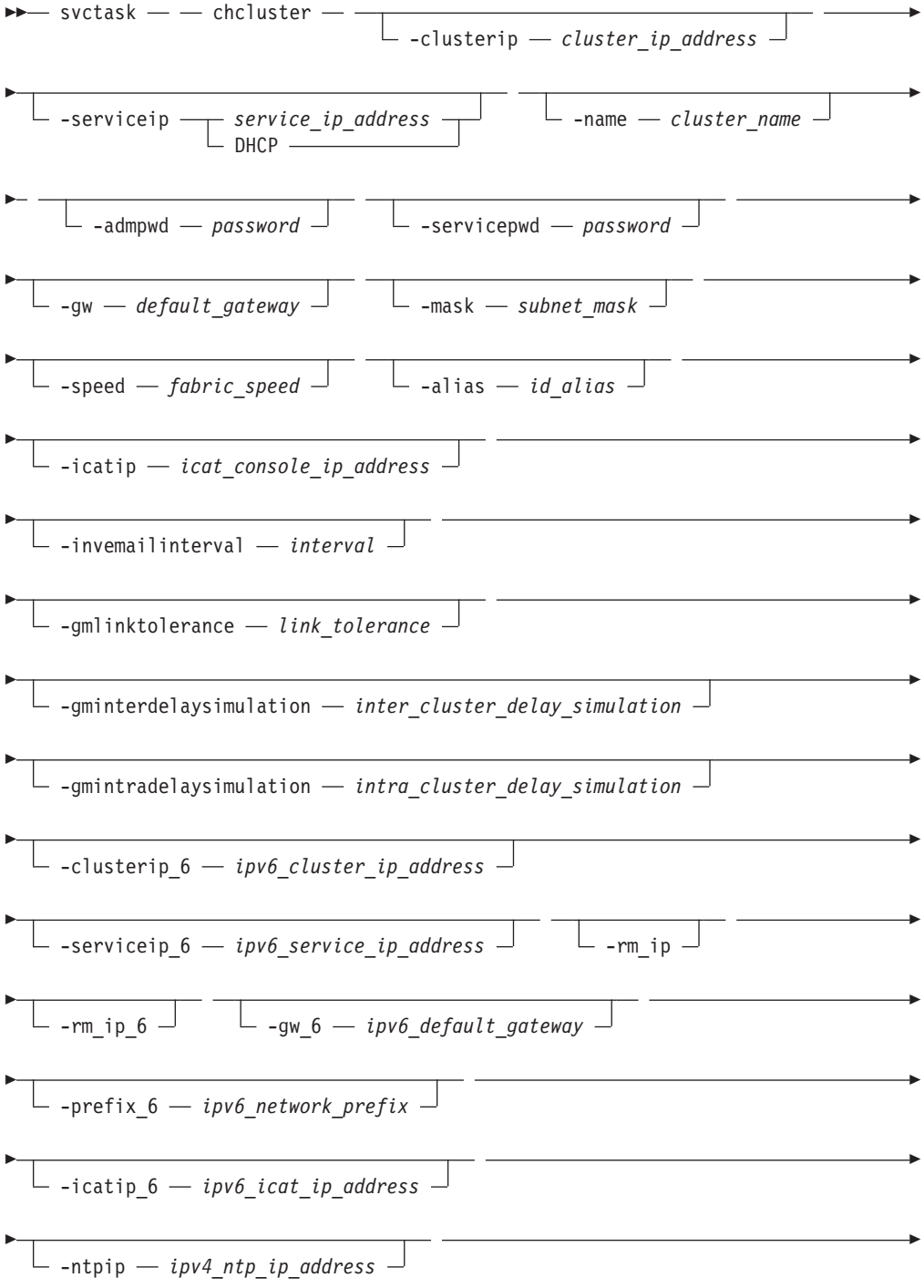
The resulting output

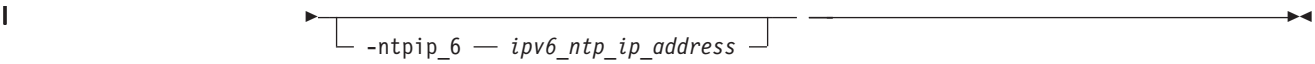
```
Node, id [6], successfully added
```

chcluster

The **chcluster** command modifies the attributes of an existing cluster. You can enter this command any time after a cluster has been created. All the parameters that are associated with this command are optional. However, you must specify one or more parameters with this command.

Syntax





Parameters

-clusterip *cluster_ip_address*
(Optional) Specifies the new cluster IP address.

Note: After the cluster IP address is changed, you lose the open shell connection to the cluster. You must reconnect with the newly specified IP address.

-serviceip *service_ip_address*
(Optional) Specifies the new service IPv4 address. This address is the address to use if the node must be started after it has been released from the cluster. Specify either a fixed IPv4 address, or to use a dynamic IP address, specify **DHCP**.

-name *cluster_name*
(Optional) Specifies a new name for the cluster.

-admpwd *password*
(Optional) Specifies a new administrator password. You can specify this parameter with or without the password. If this parameter is not followed by a password, you are prompted for the password. When you type the password in response to the prompt, the password is not displayed.

Note: Only a user with administrator authority can change the password.

-servicepwd *password*
(Optional) Specifies a new service user password. You can specify this parameter with or without the password. If the parameter is not followed by a password, you are prompted for the password. When you type the password in response to the prompt, the password is not displayed.

Note: Only a user with administrator authority can change the password.

-gw *default_gateway*
(Optional) Specifies the new default gateway IPv4 address of the cluster.

-mask *subnet_mask*
(Optional) Specifies the new IPv4 subnet mask of the cluster.

-speed *fabric_speed*
(Optional) Specifies the speed of the fabric to which this cluster is attached. Valid values are 1 or 2 (GB).

Attention: Changing the speed on a running cluster breaks I/O service to the attached hosts. Before changing the fabric speed, stop I/O from active hosts and force these hosts to flush any cached data by demounting volumes (for UNIX® host types) or by removing drive letters (for Windows host types). Some hosts might need to be rebooted to detect the new fabric speed.

The fabric speed setting applies only to the SAN Volume Controller models 2145-4F2 and 2145-8F2 in a cluster. The SAN Volume Controller models 2145-8A4, 2145-8G4, and 2145-8F4 automatically negotiate the fabric speed on a per-port basis.

-alias *id_alias*
(Optional) Specifies an alternate name that does not change the basic ID for the

cluster, but does influence the VDisk_UID of every **vdiskhostmap**, both existing and new. These objects appear to have been created for a cluster whose ID matches the alias.

-icatip *icat_console_ip_address*
(Optional) Specifies the new IP address that is used by the cluster. The format of this IP address must be a dotted decimal notation with the port; for example, 255.255.255.255:8080. If you specify this parameter, it overwrites any existing **-icatip_6** address.

-invemailinterval *interval*
(Optional) Specifies the interval at which inventory e-mails are sent to the designated e-mail recipients. The interval range is 0 to 15. The interval is measured in days. Setting the value to 0 turns the inventory e-mail notification function off.

-gmlinktolerance *link_tolerance*
(Optional) Specifies the length of time, in seconds, for which an inadequate intercluster link is tolerated for a Global Mirror operation. The parameter accepts values from 60 to 400 seconds in steps of 10 seconds. The default is 300 seconds. You can disable the link tolerance by entering a value of zero (0) for this parameter.

-gminterdelaysimulation *inter_cluster_delay_simulation*
(Optional) Specifies the intercluster delay simulation, which simulates the Global Mirror round trip delay between two clusters, in milliseconds. The default is 0; the valid range is 0 to 100 milliseconds.

-gmintradelaysimulation *intra_cluster_delay_simulation*
(Optional) Specifies the intracluster delay simulation, which simulates the Global Mirror round trip delay in milliseconds. The default is 0; the valid range is 0 to 100 milliseconds.

-clusterip_6 *ipv6_cluster_ip_address*
(Optional) Specifies the new cluster IPv6 address.

Note: After the cluster IP address is changed, you lose the open shell connection to the cluster. You must reconnect with the newly specified IP address.

-serviceip_6 *ipv6_service_ip_address*
(Optional) Specifies the service IPv6 address for the cluster. Use this address if the node must be started after it has been released from the cluster. Specify either a fixed IPv6 address, or to use a dynamic IPv6 address, specify **DHCP**.

-rm_ip
(Optional) Deletes all IPv4 addresses in the cluster.

-rm_ip_6
(Optional) Deletes all IPv6 addresses in the cluster.

-gw_6 *ipv6_default_gateway*
(Optional) Specifies the IPv6 default gateway for the cluster.

-prefix_6 *ipv6_network_prefix*
(Optional) Specifies the IPv6 network prefix for the cluster. The *ipv6_network_prefix* value is 0 - 127.

-icatip_6 *icat_console_ipv6_address*
(Optional) Specifies the new IPv6 address that is used by the cluster. If you specify this parameter, it overwrites any existing **-icatip** address. The format of the IPv6 address must be one of the following:

- Eight colon-separated groups of four hexadecimal digits; for example:
  `[1234:1234:abcd:0123:0000:0000:7689:6576]:23`
- Eight colon-separated groups of hexadecimal digits with leading zeros omitted; for example:
  `[1234:1234:abcd:123:0000:0000:7689:6576]:23`
- Suppression of one or more consecutive all 0 groups; for example:
  `[1234:1234:abcd:123::7689:6576]:23`

-ntpip *ipv4_ntp_ip_address*
(Optional) Specifies the IPv4 address for the Network Time Protocol (NTP) server. Configuring an NTP server address causes the cluster to immediately start using that NTP server as its time source. To stop using the NTP server as a time source, invoke the **-ntpip** parameter with a zero address, as follows:

```
svctask chcluster -ntpip 0.0.0.0
```

-ntpip_6 *ipv6_ntp_ip_address*
(Optional) Specifies the IPv6 address for the NTP server. Configuring an NTP server address causes the cluster to immediately start using that NTP server as its time source. To stop using the NTP server as a time source, invoke the **-ntpip_6** parameter with a zero address, as follows:

```
svctask chcluster -ntpip_6 0::0
```

Description

This command modifies specific features of a cluster. Multiple features can be changed by issuing a single command.

If the cluster IP address is changed, the open command-line shell closes during the processing of the command. You must reconnect to the new IP address.

The service IP address is not used until a node is expelled from the cluster. If this node cannot rejoin the cluster, you can bring the node up in service mode. In this mode, the node can be accessed as a stand-alone node using the service IP address.

Using the **-ntpip** or **-ntpip_6** parameter allows the cluster to use an NTP server as an outside time source. The cluster adjusts the system clock of the configuration node according to time values from the NTP server. The clocks of the other nodes are updated from the configuration node's clock. In the NTP mode, the **svctask setclustertime** command is disabled.

All command parameters are optional; however, you must specify at least one parameter.

Modifying a password: To change the administrator user password, issue the **svtask chcluster -admpwd** ***password*** command. To change the service user password, issue the **svtask chcluster -servicepwd** ***password*** command.

Note: If you do not want the password to display as you enter it on the command line, omit the new password. The command line tool then prompts you to enter and confirm the password without the password being displayed.

Modifying an IP address: List the IP address of the cluster by issuing the **svcinfo lscluster** command. Modify the IP address by issuing the **svctask chcluster** command. You can either specify a static IP address or have the system assign a dynamic IP address.

Table 1 provides IP address formats that are supported.

Table 1. ip_address_list formats. Supported IP address formats

IP type	*ip_address_list* **format**
IPv4 (no port set, SVC uses default)	`1.2.3.4`
IPv4 with specific port	`1.2.3.4:22`
Full IPv6, default port	`1234:1234:0001:0123:1234:1234:1234:1234`
Full IPv6, default port, leading zeros suppressed	`1234:1234:1:123:1234:1234:1234:1234`
Full IPv6 with port	`[2002:914:fc12:848:209:6bff:fe8c:4ff6]:23`
Zero-compressed IPv6, default port	`2002::4ff6`
Zero-compressed IPv6 with port	`[2002::4ff6]:23`

An invocation example

```
svctask chcluster -clusterip 9.20.165.16 -servicepwd myownpassw0rd -gw 9.13.56.87
```

The resulting output

```
No feedback
```

An IPv6 invocation example

```
svctask chcluster -clusterip_6 2002:914:fc12:848:209:6bff:fe8c:4ff6
-prefix_6 64 -gw_6 fe80::7:1234:5678:9abc
```

The resulting output

```
No feedback
```

A mixed IPv4 and IPv6 invocation example

```
svctask chcluster -clusterip 9.20.165.16 -servicepwd myownpassw0rd -gw 9.13.56.87
-clusterip_6 2002:914:fc12:848:209:6bff:fe8c:4ff6 -prefix_6 64
-gw_6 fe80::7:1234:5678:9abc
```

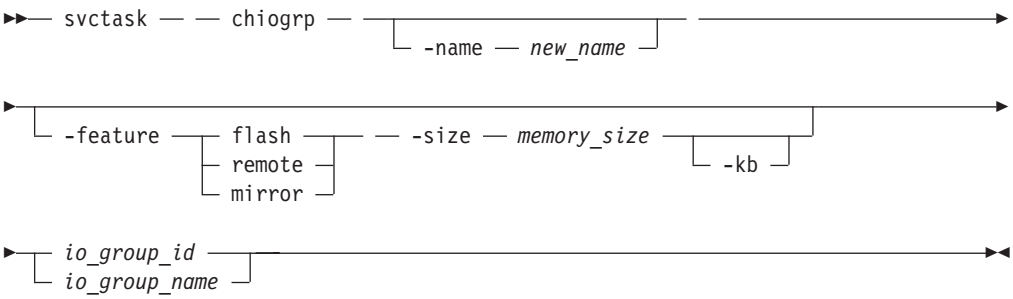
The resulting output

```
No feedback
```

chiogrp

The **chiogrp** command modifies the name of an I/O group, or the amount of memory that is available for Copy Services or VDisk mirroring operations.

Syntax



Parameters

-name *new_name*
(Optional) Specifies the name to assign to the I/O group. The **-name** parameter cannot be specified with the **-feature**, **-size**, or **-kb** parameters.

-feature flash | remote | mirror
(Optional) Specifies the feature to modify the amount of memory for: Copy Services or VDisk mirroring. You must specify this parameter with the **-size** parameter. You cannot specify this parameter with the **-name** parameter.

Note: Specifying **remote** changes the amount of memory that is available for Metro Mirror or Global Mirror processing. Any VDisk that is in a Metro Mirror or Global Mirror relationship uses memory in its IO group, including master and auxiliary VDisks, and VDisks that are in inter-cluster or intra-cluster relationships.

-size *memory_size*
(Optional) Specifies the amount of memory that is available for the specified Copy Services or VDisk mirroring function. Valid input is **0** or any integer. The default unit of measurement for this parameter is megabytes (MB); you can use the kilobytes **-kb** parameter to override the default. You must specify this parameter with the **-feature** parameter. You cannot specify this parameter with the **-name** parameter.

-kb
(Optional) Changes the units for the **-size** parameter from megabytes (MB) to kilobytes (KB). If you specify this parameter, the **-size** *memory_size* value must be any number divisible by 4. You must specify this parameter with the **-feature** and **-size** parameters. You cannot specify this parameter with the **-name** parameter.

io_group_id | io_group_name
(Required) Specifies the I/O group to modify. You can modify an I/O group by using the **-name** or the **-feature** parameter.

Description

The **chiogrp** command modifies the name of an I/O group or the amount of memory that is available for Copy Services or VDisk mirroring. You can assign a name to an I/O group or change the name of a specified I/O group. You can change the amount of memory that is available for Copy Services or VDisk mirroring operations by specifying the **-feature flash | remote | mirror** parameter, and a memory size. For VDisk mirroring and Copy Services (FlashCopy, Metro Mirror, and Global Mirror), memory is traded against memory that is available to the cache. The amount of memory can be decreased or increased. Consider the following memory sizes when you use this command:

- The default memory size for FlashCopy is 20 MB.
- The default memory size for Metro Mirror and Global Mirror is 20 MB.
- The default memory size for mirrored VDisks is 0 MB.
- The maximum memory size that can be specified for FlashCopy is 512 MB.
- The maximum memory size that can be specified for Metro Mirror and Global Mirror is 512 MB.
- The maximum memory size that can be specified for mirrored VDisks is 512 MB.
- The maximum combined memory size across all features is 512 MB.

Table 2 demonstrates the amount of memory required for VDisk mirroring and Copy Services. Each 1 MB of memory provides the following VDisk capacities and grain sizes:

Table 2. Memory required for VDisk Mirroring and Copy Services

Feature	Grain size	1 MB of memory provides the following VDisk capacity for the specified I/O group
Metro Mirror and Global Mirror	256 KB	2 TB of total Metro Mirror and Global Mirror VDisk capacity
FlashCopy	256 KB	2 TB of total FlashCopy source VDisk capacity
FlashCopy	64 KB	512 GB of total FlashCopy source VDisk capacity
Incremental FlashCopy	256 KB	1 TB of total Incremental FlashCopy source VDisk capacity
Incremental FlashCopy	64 KB	256 GB of total Incremental FlashCopy source VDisk capacity
VDisk mirroring	256 KB	2 TB of mirrored VDisks

For multiple FlashCopy targets, you must consider the number of mappings. For example, for a mapping with a 256 KB grain size, 8 KB of memory allows one mapping between a 16 GB source VDisk and a 16 GB target VDisk. Alternatively, for a mapping with a 256 KB grain size, 8 KB of memory allows two mappings between one 8 GB source VDisk and two 8 GB target VDisks.

When you create a FlashCopy mapping, if you specify an I/O group other than the I/O group of the source VDisk, the memory accounting goes towards the specified I/O group, not towards the I/O group of the source VDisk.

An invocation example

```
svctask chiogrp -name testiogrpone io_grp0
```

The resulting output

```
No feedback
```

An invocation example for changing the amount of FlashCopy memory in io_grp0 to 30 MB

```
svctask chiogrp -feature flash -size 30 io_grp0
```

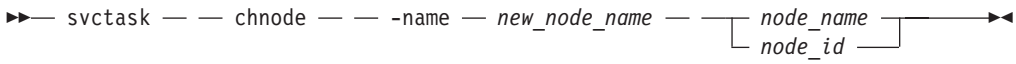
The resulting output

```
No feedback
```

chnode

You can use the **chnode** command to change the name that is assigned to a node. The name can then be used when running subsequent commands.

Syntax



Parameters

-name *new_node_name*
Specifies the name to assign to the node.

node_name | *node_id*
Specifies the node to be modified. The variable that follows the parameter is either:

- The node name that you assigned when you added the node to the cluster.
- The node ID that is assigned to the node (not the worldwide node name).

Description

This command changes the name of the node in question. The node name can then be used when running subsequent commands.

You can use the **chnode** command to change the name that is assigned to a node. The name can then be used when running subsequent commands.

An invocation example

```
svctask chnode -name testnodeone nodeone
```

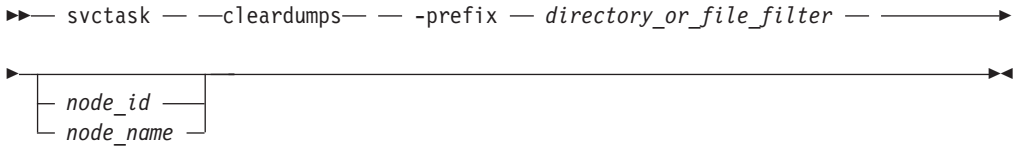
The resulting output

```
No feedback
```

cleardumps

The **cleardumps** command cleans the various dump directories on a specified node.

Syntax



Parameters

-prefix *directory_or_file_filter*
(Required) Specifies the directory, files, or both to be cleaned. If a directory is specified, with no file filter, all relevant dump or log files in that directory are cleaned. You can use the following directory arguments (filters):

- **/dumps** (cleans all files in all subdirectories)
- **/dumps/cimom**
- **/dumps/configs**
- **/dumps/elogs**
- **/dumps/feature**

- **/dumps/iostats**
- **/dumps/iotrace**
- **/home/admin/upgrade**

In addition to the directory, you can specify a filter file. For example, if you specify `/dumps/elogs/*.txt`, all files in the **/dumps/elogs** directory that end in `.txt` are cleaned.

Note: The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard.
- With a wildcard, you must use double quotation marks (″ ″) around the filter entry, such as in the following entry:

```
>svctask cleardumps -prefix "/dumps/elogs/*.txt"
```

node_id | node_name

(Optional) Specifies the node to be cleaned. The variable that follows the parameter is either:

- The node name, that is, the label that you assigned when you added the node to the cluster
- The node ID that is assigned to the node (not the worldwide node name).

Description

This command deletes all the files that match the directory/file_filter argument on the specified node. If no node is specified, the configuration node is cleaned.

You can clean all the dumps directories by specifying /dumps as the directory variable.

You can clean all the files in a single directory by specifying one of the directory variables.

You can list the contents of these directories on the given node by using the **svcinfo lsxxxxdumps** commands.

You can use this command to clean specific files in a given directory by specifying a directory or file name. You can use the wildcard character as part of the file name.

Note: To preserve the configuration and trace files, any files that match the following wildcard patterns are not cleaned:

- *svc.config*
- *.trc
- *.trc.old

An invocation example

```
svctask cleardumps -prefix /dumps/configs
```

The resulting output

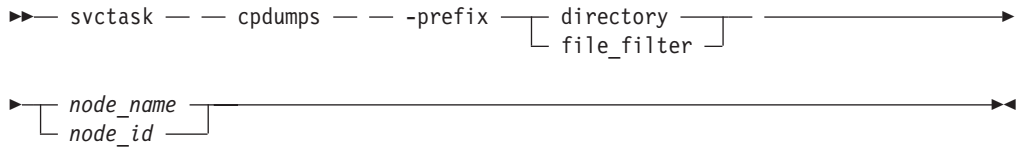
```
No feedback
```

cpdumps

The **cpdumps** command copies dump files from a nonconfiguration node onto the configuration node.

Note: In the rare event that the /dumps directory on the configuration node is full, the copy action ends when the directory is full and provides no indicator of a failure. Therefore, clear the /dumps directory after migrating data from the configuration node.

Syntax



Parameters

-prefix *directory* | *file_filter*
(Required) Specifies the directory, or files, or both to be retrieved. If a directory is specified with no file filter, all relevant dump or log files in that directory are retrieved. You can use the following directory arguments (filters):

- **/dumps** (retrieves all files in all subdirectories)
- **/dumps/audit**
- **/dumps/cimom**
- **/dumps/configs**
- **/dumps/elogs**
- **/dumps/feature**
- **/dumps/iostats**
- **/dumps/iotrace**
- **/home/admin/upgrade**

In addition to the directory, you can specify a file filter. For example, if you specified `/dumps/elogs/*.txt`, all files in the **/dumps/elogs** directory that end in `.txt` are copied.

Note: The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard.
- When you use a wildcard, you must surround the filter entry with double quotation marks (""), as follows:

```
>svctask cleardumps -prefix "/dumps/elogs/*.txt"
```

node_id | *node_name*
(Required) Specifies the node from which to retrieve the dumps. The variable that follows the parameter can be one of the following:

- The node name, or label that you assigned when you added the node to the cluster
- The node ID that is assigned to the node (not the worldwide node name).

If the node specified is the current configuration node, no file is copied.

Description

This command copies any dumps that match the directory or file criteria from the given node to the current configuration node.

You can retrieve dumps that were saved to an old configuration node. During failover processing from the old configuration node to another node, the dumps that were on the old configuration node are not automatically copied. Because access from the CLI is only provided to the configuration node, cluster files can only be copied from the configuration node. This command enables you to retrieve files and place them on the configuration node so that you can then copy them.

You can view the contents of the directories by using the svcinfo lsxxxxdumps commands.

An invocation example

```
svctask cpdumps -prefix /dumps/configs nodeone
```

The resulting output

```
No feedback
```

detectmdisk

The **detectmdisk** command allows you to manually rescan the fibre-channel network for any new managed disks (MDisks) that might have been added and to rebalance MDisk access across all available controller device ports.

Syntax

Description

This command causes the cluster to rescan the fibre-channel network. The rescan discovers any new MDisks that have been added to the cluster and rebalances MDisk access across the available controller device ports. This command also detects any loss of controller port availability, and updates the SAN Volume Controller configuration to reflect any changes.

Note: Although it might appear that the **detectmdisk** command has completed, some extra time might be required for it to run. The **detectmdisk** is asynchronous and returns a prompt while the command continues to run in the background. You can use the **lsdiscoverystatus** command to list the discovery status.

In general, the cluster automatically detects disks when they appear on the network. However, some fibre-channel controllers do not send the required SCSI primitives that are necessary to automatically discover the new disks.

If you have attached new storage and the cluster has not detected it, you might need to run this command before the cluster detects the new disks.

When back-end controllers are added to the fibre-channel SAN and are included in the same switch zone as a cluster, the cluster automatically discovers the back-end controller and determines what storage is presented to it. The SCSI LUs that are presented by the back-end controller are displayed as unmanaged MDisks. However, if the configuration of the back-end controller is modified after this has occurred, the cluster might be unaware of these configuration changes. Run this command to rescan the fibre-channel network and update the list of unmanaged MDisks.

Note: The automatic discovery that is performed by the cluster does not write to an unmanaged MDisk. Only when you add an MDisk to an MDisk group, or use an MDisk to create an image mode virtual disk, is the storage actually used.

To identify the available MDisks, issue the **svctask detectmdisk** command to scan the fibre-channel network for any MDisks. When the detection is complete, issue the **svcinfo lsmdiskcandidate** command to show the unmanaged MDisks; these MDisks have not been assigned to an MDisk group. Alternatively, you can issue the **svcinfo lsmdisk** command to view all of the MDisks.

If disk controller ports have been removed as part of a reconfiguration, the SAN Volume Controller detects this change and reports the following error because it cannot distinguish an intentional reconfiguration from a port failure:

```
1630 Number of device logins reduced
```

If the error persists and redundancy has been compromised, the following more serious error is reported:

```
1627 Insufficient redundancy in disk controller connectivity
```

You must issue the **svctask detectmdisk** command to force SAN Volume Controller to update its configuration and accept the changes to the controller ports.

Note: Only issue the **svctask detectmdisk** command when all of the disk controller ports are working and correctly configured in the controller and the SAN zoning. Failure to do this could result in errors not being reported.

An invocation example

```
svctask detectmdisk
```

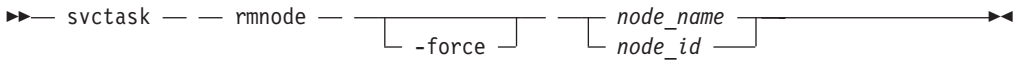
The resulting output

```
No feedback
```

rmnode

The **rmnode** command deletes a node from the cluster. You can enter this command any time after a cluster has been created.

Syntax



Parameters

-force
(Optional) Specifies that the node is forcefully deleted from the cluster without ensuring no loss of access. The cache is not flushed from the deleted node.

Attention: When the **-force** option is used, the write cache for that node is discarded rather than flushed, which can result in data loss.

node_name | node_id
Specifies the node to be deleted. The value for this parameter can be one of the following:

- The node name that you assigned when you added the node to the cluster
- The node ID that is assigned to the node (not the worldwide node name).

Description

This command removes a node from the cluster. This makes the node a candidate to be added back into this cluster or into another cluster. After the node is deleted, the other node in the I/O group enters write-through mode until another node is added back into the I/O group.

By default, the **rmnode** command flushes the cache on the specified node before the node is taken offline. In some circumstances, such as when the system is already degraded (for example, when both nodes in the I/O group are online and the virtual disks within the I/O group are degraded), the system ensures that data loss does not occur as a result of deleting the only node with the cache data.

The cache is flushed before the node is deleted to prevent data loss if a failure occurs on the other node in the I/O group.

To take the specified node offline immediately without flushing the cache or ensuring data loss does not occur, run the **rmnode** command with the **-force** parameter.

Prerequisities:

Before you issue the **rmnode** command, perform the following tasks and read the following Attention notices to avoid losing access to data:

1. Determine which virtual disks (VDisks) are still assigned to this I/O group by issuing the following command. The command requests a filtered view of the VDisks, where the filter attribute is the I/O group.

   ```
   svcinfo lsvdisk -filtervalue IO_group_name=name
   ```

   where *name* is the name of the I/O group.

   Note: Any VDisks that are assigned to the I/O group that this node belongs to are assigned to the other node in the I/O group; the preferred node is changed. You cannot change this setting back.
2. Determine the hosts that the VDisks are mapped to by issuing the **svcinfo lsvdiskhostmap** command.
3. Determine if any of the VDisks that are assigned to this I/O group contain data that you need to access:
   - If you *do not* want to maintain access to these VDisks, go to step 5 on page 51.

- If you *do* want to maintain access to some or all of the VDisks, back up the data or migrate the data to a different (online) I/O group.

4. Determine if you need to turn the power off to the node:
   - If this is the last node in the cluster, you do not need to turn the power off to the node. Go to step 5.
   - If this is *not* the last node in the cluster, turn the power off to the node that you intend to remove. This step ensures that the Subsystem Device Driver (SDD) does not rediscover the paths that are manually removed before you issue the delete node request.
5. Update the SDD configuration for each virtual path (vpath) that is presented by the VDisks that you intend to remove. Updating the SDD configuration removes the vpaths from the VDisks. Failure to update the configuration can result in data corruption. See the *Multipath Subsystem Device Driver: User's Guide* for details about how to dynamically reconfigure SDD for the given host operating system.
6. Quiesce all I/O operations that are destined for the node that you are deleting. Failure to quiesce the operations can result in failed I/O operations being reported to your host operating systems.

Attention:

1. Removing the last node in the cluster destroys the cluster. Before you delete the last node in the cluster, ensure that you want to destroy the cluster.
2. If you are removing a single node and the remaining node in the I/O group is online, the data can be exposed to a single point of failure if the remaining node fails.
3. This command might take some time to complete since the cache in the I/O group for that node is flushed before the node is removed. If the **-force** parameter is used, the cache is not flushed and the command completes more quickly. However, if the deleted node is the last node in the I/O group, using the **-force** option results in the write cache for that node being discarded rather than flushed, and data loss can occur. The **-force** option should be used with caution.
4. If both nodes in the I/O group are online and the VDisks are already degraded before deleting the node, redundancy to the VDisks is already degraded and loss of access to data and loss of data might occur if the **-force** option is used.

Notes:

1. If you are removing the configuration node, the **rmnode** command causes the configuration node to move to a different node within the cluster. This process might take a short time: typically less than a minute. The cluster IP address remains unchanged, but any SSH client attached to the configuration node might need to reestablish a connection. The SAN Volume Controller Console reattaches to the new configuration node transparently.
2. If this is the last node in the cluster or if it is currently assigned as the configuration node, all connections to the cluster are lost. The user interface and any open CLI sessions are lost if the last node in the cluster is deleted. A time-out might occur if a command cannot be completed before the node is deleted.

An invocation example

```
svctask rmnode 1
```

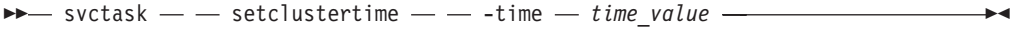
The resulting output

```
No feedback
```

setclustertime

The **setclustertime** command allows you to set the time for the cluster.

Syntax



Parameters

-time *time_value*
(Required) Specifies the time to which the cluster must be set. This must be in the following format:

MMDDHHmmYYYY

Description

This command sets the time for the cluster.

An invocation example

```
svctask setclustertime -time 040509142003
```

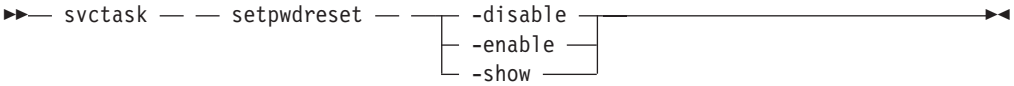
The resulting output

```
No feedback
```

setpwdreset

Use the **setpwdreset** command to view and change the status of the password-reset feature for the display panel.

Syntax



Parameters

-disable
Disables the password-reset feature that is available through the front panel menu system.

-enable
Enables the password-reset feature that is available through the front panel menu system.

-show
Displays the status of the password-reset feature, which is either `enabled` or `disabled`.

Description

The front panel menu system provides an option to reset the administrator password. This option resets the password to a random string that is displayed on the front panel. You can then use this password to access the system. You can change the password at the next login.

Issue the **svctask setpwdreset** command to view and change the status of the password-reset feature for the display panel. Passwords can consist of the following characters: A - Z, a - z, 0 - 9, and underscore (_). Make a careful note of the admin password, because without it, you cannot access the cluster.

This command allows you access in case the administrator password is forgotten. If you leave this feature enabled, you can ensure adequate physical security to the cluster hardware.

You can view or change the status of this feature.

An invocation example

```
svctask setpwdreset -show
```

The resulting output

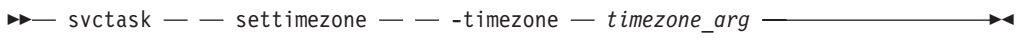
```
Password status: [1]
```

This output means that the password or reset feature that is available through the front panel menu system is enabled. If the password status is [0], this feature is disabled.

settimezone

Use the **settimezone** command to set the time zone for the cluster.

Syntax



Parameters

-timezone *timezone_arg*
Specifies the time zone to set for the cluster.

Description

This command sets the time zone for the cluster. Use the **-timezone** parameter to specify the numeric ID of the time zone that you want to set. Issue the **svcinfo lstimezones** command to list the time-zones that are available on the cluster. A list of valid time-zones settings are displayed in a list.

The time zone that this command sets will be used when formatting the error log that is produced by issuing the following command:

```
svctask dumperrlog
```

Note: If you have changed the timezone, you must clear the error log dump directory before you can view the error log through the Web application.

Issue the **svcinfo showtimezone** command to display the current time-zone settings for the cluster. The cluster ID and its associated time-zone are displayed. Issue the **svctask setclustertime** command to set the time for the cluster.

An invocation example

```
svctask settimezone -timezone 5
```

The resulting output

```
No feedback
```

startstats

Use the **startstats** command to start the collection of statistics for both virtual disks (VDisks) and managed disks (MDisks).

Syntax

Parameters

-interval *time_in_minutes*
Specifies the time in minutes. This is the time interval between the gathering of statistics, between 1 and 60 minutes in increments of 1.

Description

Statistics are collected at the end of each sampling period (as specified by the **-interval** parameter). These statistics are written to a file. A new file is created at the end of each sampling period. Separate files are created for MDisks, VDisks and node statistics.

The files generated are written to the /dumps/iostats directory.

A maximum of 16 files are stored in the directory at any one time for each statistics file type, for example:

```
Nm_stats_nodepanelname_date_time
Nv_stats_nodepanelname_date_time
Nn_stats_nodepanelname_date_time
m_stats_nodepanelname_date_time
v_stats_nodepanelname_date_time
```

Statistics files beginning with m_stats_* and v_stats_* are not created if the specified time interval is less than 15 minutes. Statistics files beginning with Nm_stats_*, Nv_stats_* and Nn_stats_* are created for all time intervals.

Before the 17th file (for each type) is created, the oldest file of that type is deleted.

These files can be listed by using the **svcinfo lsiostatsdumps** command.

The following naming convention is used for these files:

*stats_type*_stats_*nodepanelname_date_time*

Where *stats_type* is m or Nm for MDisks, v or Nv for VDisks, and Nn for node statistics. *nodepanelname* is the current configuration node panel name, *date* is in the format of yymmdd, and *time* is in the format of hhmmss.

The following are examples of MDisk file names:

```
m_stats_000229_031123_072426
Nm_stats_000229_031123_072426
```

The following are examples of VDisk file names:

```
v_stats_000229_031123_072426
Nv_stats_000229_031123_072426
```

The following is an example of a node statistics file name:

```
Nn_stats_000229_031123_072426
```

The statistics that are collected for each MDisk and VDisk are reported in the m_stats_*nodepanelname_date_time* and v_stats_*nodepanelname_date_time* files and include the following statistical information:

- The number of SCSI read and write commands that are processed during the sample period
- The number of blocks of data that are read and written during the sample period

Statistics are collected for each MDisk and recorded in the Nm_stats_*nodepanelname_date_time* file, including the following statistical information:

- The number of SCSI read and write commands that are processed during the sample period
- The number of blocks of data that are read and written during the sample period
- Per MDisk, cumulative read and write external response times in milliseconds
- Per MDisk, cumulative read and write queued response times

Statistics are collected for each VDisk and recorded in the Nv_stats_*nodepanelname_date_time* file, including the following statistical information:

- The total number of processed SCSI read and write commands
- The total amount of read and written data
- Cumulative read and write response time in milliseconds
- Statistical information about the read/write cache usage
- Global Mirror statistics including latency

Statistics are collected for the node from which the statistics file originated and recorded in the Nn_stats_*nodepanelname_date_time* file, including the following statistical information:

- Usage figure for the node from which the statistic file was obtained
- The amount of data transferred to and received from each port on the node to other devices on the SAN
- Statistical information about communication to other nodes on the fabric

Note: The v_* and m_* statistics are per-cluster statistics that are only collected on the configuration node. The Nm_*, Nn_* and Nv_* files are per-node statistics that are generated on each node.

An invocation example

```
svctask startstats -interval 25
```

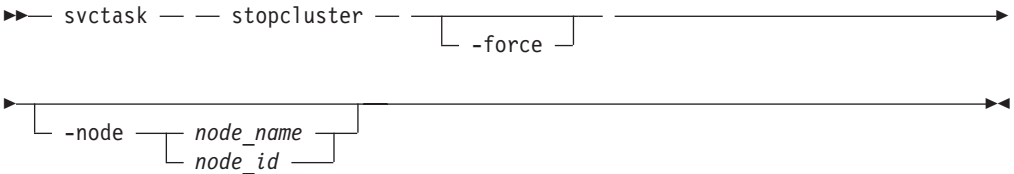
The resulting output

```
No feedback
```

stopcluster

The **stopcluster** command allows you to shut down a single node or the entire cluster in a controlled manner. When you issue this command, you are prompted with a confirmation of intent to process the command.

Syntax



Parameters

-force
(Optional) Specifies that the node that is being shut down is the last online node in a given I/O group.

-node *node_name | node_id*
(Optional) Specifies the node that you want to shut down. You can specify one of the following values:

- The node name, or label that you assigned when you added the node to the cluster
- The node ID that is assigned to the node (not the worldwide node name).

If you specify **-node** *node_name | node_id*, only the specified node is shut down; otherwise, the entire cluster is shut down.

Description

When you enter this command with no parameters, the entire cluster is shutdown. All data is flushed to disk before the power is removed.

Attention: Ensure that you have stopped all FlashCopy, Metro or Global Mirror, or data migration operations before you attempt to shutdown a node or cluster. You must also ensure that all asynchronous deletion operations have completed prior to a shutdown operation.

When you enter this command with either a node ID or node name, the node in question is shut down. After the command completes, the remaining node in the I/O group enters write-through mode until the power to the node is returned and the node rejoins the cluster.

Attention: If you have to remove all cluster input power for more than a few minutes, you must shut down the cluster before you remove the power. If the input power is removed from the uninterruptible power supply units without first shutting down the cluster and the uninterruptible power supply units, the uninterruptible power supply units remain operational and eventually become drained of power.

When input power is restored to the uninterruptible power supply units, they begin to recharge but the nodes do not permit I/O activity on the virtual disks until the uninterruptible power supply is charged enough to enable all the data on the nodes to be saved in the event of an unexpected power loss. This might take as long as two hours. Shutting down the cluster prior to removing input power to the uninterruptible power supply units prevents the draining of battery power and makes it possible for I/O activity to resume as soon as input power is restored.

Attention: Before shutting down a node or the cluster, quiesce all I/O operations that are destined for this node or cluster. Failure to quiesce can result in failed I/O operations being reported to your host operating systems.

Begin the process of quiescing all I/O to the cluster by stopping the applications on your hosts that are using the VDisks that are provided by the cluster. Perform the following steps to quiesce all I/O to the cluster:

1. If you are unsure which hosts are using the VDisks that are provided by the cluster, determine the hosts that the VDisks are mapped to, as follows:
   a. List the hosts that this VDisk is mapped to by issuing the following command, substituting the *vdiskname | id* variable with the name or ID of the VDisk:

      svcinfo lsvdiskhostmap *vdiskname | id*

   b. In the command output, located the host name or ID to determine which host the specified VDisk is mapped to. If no data is displayed, the VDisk is not mapped to any hosts.
2. Repeat the previous procedure for all VDisks.

Attention: If you are shutting down the entire cluster, you lose access to all VDisks that are being provided by this cluster.

When all I/O has been stopped, issue the **svctask stopcluster** to shut down a single node or the entire cluster in a controller manner. If you specify the node ID or node name, you can shut down a single node. After the command completes, the other node in the I/O group goes into write-through mode until the power to the node is returned and the node rejoins the cluster.

Attention: If this is the last node in an I/O group, you must specify the **-force** parameter. You will lose all access to the virtual disks in the I/O group. Before you enter this command, ensure that this is what you want to do.

If a shutdown command has been sent to the cluster and both cluster and uninterruptible power supply units have powered off, when input power is restored, press the power button on the uninterruptible power supply front panel to restart the uninterruptible power supply units.

Ensure that you have stopped all FlashCopy mappings and Metro or Global Mirror relationships. In addition, ensure that all data migration operations and forced deletions have completed before continuing. Entering `y` to the confirmation

message processes the command. `No feedback` is then displayed. Entering anything other than `y` or `Y` results in the command not processing. `No feedback` is displayed.

Attention: If you are shutting down a single node and the other node in the I/O group is online, the cache on the partner node goes into the write-through mode and that you are exposed to a single point of failure if the partner node fails while the node is shut down. You also lose access to all VDisks being served by this I/O group.

An invocation example

```
svctask stopcluster
```

The resulting output You will be presented with the following warning:

```
Are you sure that you want to continue with the shut down?
```

stopstats

You can use the **stopstats** command to stop the collection of statistics for both VDisks and MDisks.

Syntax

Description

This command turns off the generation of statistics, until you start them again (with the **svctask startstats**).

An invocation example

```
svctask stopstats
```

The resulting output

```
No feedback
```

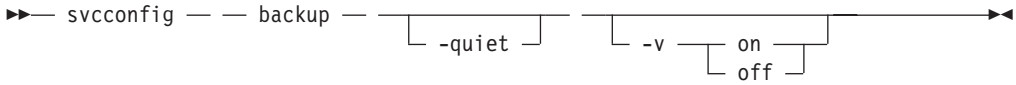
Chapter 8. Backup and restore commands

The following commands are used for backing up and restoring configuration information with the SAN Volume Controller.

backup

Use the **backup** command to back up your configuration. You can enter this command any time after a cluster has been created.

Syntax



Parameters

-quiet
Suppresses standard output (STDOUT) messages from the console.

-v on | off
On means verbose messages are displayed. Off means normal messages (the default) are displayed.

Description

The **backup** command extracts configuration information from the cluster, allowing you to restore your configuration whenever necessary. The **backup** command produces **.xml**, **.sh**, **.log**, and **.key** files and saves them in the **/tmp** directory. The **.xml** file contains the extracted configuration information. The **.log** file contains details about command usage.

Note: If a previous **svc.config.backup.xml** file exists in **/tmp**, it is archived as **svc.config.backup.bak**; only one archive file is stored in the **/tmp** directory. Immediately archive the **.xml** file and the related **.key** files, and use the **clear** command to erase those files from the **/tmp** directory. Change all objects with default names to nondefault names; you cannot restore objects with default names.

The underscore character (_) prefix is reserved for backup and restore command usage; do not use the underscore character in any object names.

Note the following **backup** command limitation: **.key** SSH public key value files are not produced with the **.xml** file in **/tmp**. Warnings are nevertheless issued for the missing file or files that must be supplied. These files conform to the template **svc.config**.*identifier*.*user*.**key** where *identifier* and *user* are as specified for the **addsshkey** command. You are requested to supply these files if they are used with the **addsshkey** command. If they are not available, you must install a new set of keys during any future cluster restoration process.

Note: The following output is an example of a message that is displayed during the backup process if a **.key** file is not found. In this case, the key cannot be restored if a T3 or T4 recovery is run; you must restore the key manually.

```
CMMVC6136W No SSH key file svc.config.admin_crumpler.admin.key
```

An invocation example

```
svcconfig backup
```

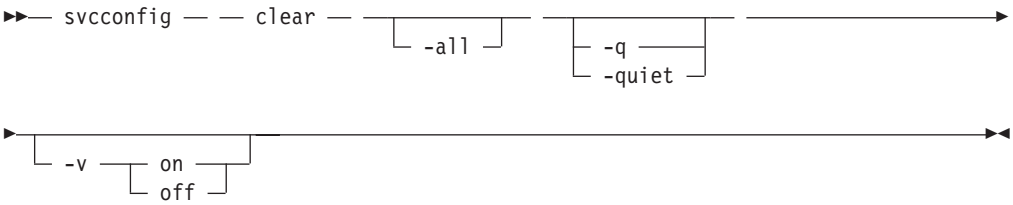
The resulting output

```
No feedback
```

clear

Use the **clear** command to erase files in the **/tmp** directory that were previously produced by other **svcconfig** commands. You can enter this command any time after a cluster has been created.

Syntax



Parameters

-all
Erases all configuration files.

-q | quiet
Suppresses console output (STDOUT).

-v on | off
Produces verbose output (on); the default is regular output (off).

Description

This command erases configuration files on the current config node.

You can use the **svcconfig clear** command without the **-all** parameter to erase files of the form:

```
/tmp/svc.config*.sh
/tmp/svc.config*.log
```

You can use the **svcconfig clear** command with the **-all** parameter to erase files of the form:

```
/tmp/svc.config*.sh
/tmp/svc.config*.log
/tmp/svc.config*.xml
/tmp/svc.config*.bak
/tmp/svc.config*.key
```

An invocation example

```
svcconfig clear -all
```

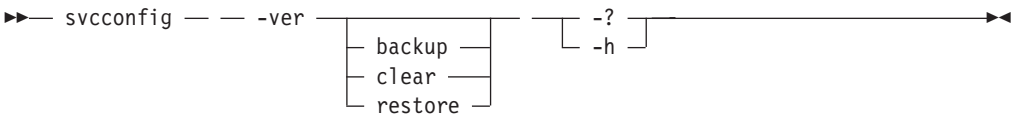
The resulting output

```
No feedback
```

help

Use the **help** command to obtain summary information about the syntax of the **svcconfig** command. You can enter this command any time after a cluster has been created.

Syntax



Parameters

-ver
Returns the version number for the **svcconfig** command.

(action) -h | -?
Provides command help: the possible values for (action) are backup, clear, and restore.

-h | -?
Provides general help.

Description

This command provides syntax help for svcconfig.

An invocation example

```
svcconfig -ver
svcconfig -?
svcconfig backup -h
```

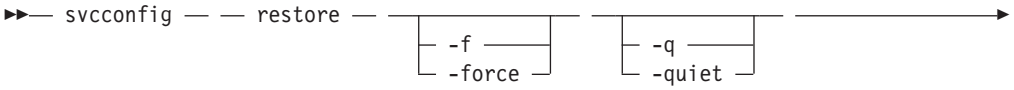
The resulting output

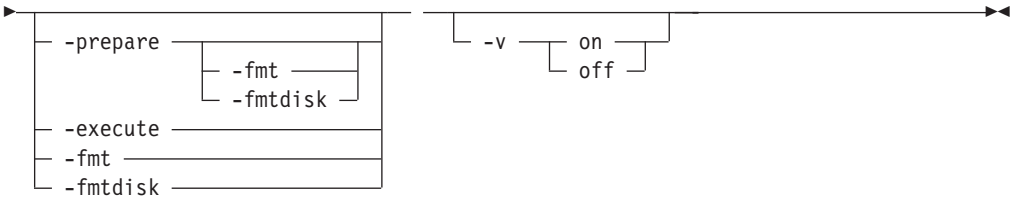
The help text displays.

restore

The **restore** command uses the configuration files in the **/tmp** directory to restore the cluster to its previous configuration.

Syntax





Parameters

-f | force
Forces continued processing where possible.

-q | quiet
Suppresses console output (STDOUT).

-prepare
Checks the current configuration against the information in **svc.config.backup.xml** on the configuration to be restored. Prepares commands for processing in **svc.config.restore.sh**, and produces a log of events in **svc.config.restore.prepare.log**.

-fmt | fmtdisk
Includes the **-fmtdisk** option an all **mkvdisk** commands to be issued.

-execute
Runs the command script **svc.config.restore.sh**. Produces a log of events in **svc.config.restore.execute.log**.

-v on | off
Produces verbose output (on); the default is regular output (off).

Description

The **restore** command restores the target cluster configuration from the **svc.config.backup.xml** file, and associated **.key** files (if present) in the configuration files directory. If neither the **-prepare** nor **-execute** option is specified, only a single event log **svc.config.restore.log** is produced.

The command pauses for 5 minutes if any nodes are added during this process. You are informed of this at run-time.

After restoration, a VDisk consists of a specific list of MDisks. If the relevant MDisk group comprises a larger list, either now or in the future, the restored VDisk cannot manage any MDisks that are not presently in its own list.

The configuration files directory is **/tmp**.

An invocation example

```
svcconfig restore -prepare
svcconfig restore -execute
```

The resulting output

```
No feedback
```

Chapter 9. Cluster diagnostic and service-aid commands

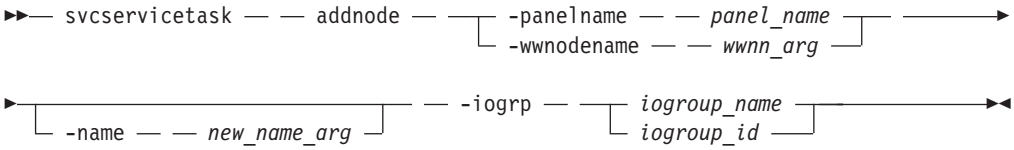
Cluster diagnostic and service-aid commands are designed to diagnose and find cluster problems.

The SAN Volume Controller enables you to perform service activity, such as problem determination and repair activities, with a limited set of command-line tools. When you are logged in under the administrator role, all command-line activities are permitted. When you are logged in under the service role, only those commands that are required for service are enabled. All of these commands apply under the service role.

addnode

The **addnode** command allows you to add a new (candidate) node to an existing cluster. You can enter this command any time after a cluster has been created.

Syntax



Parameters

-panelname *panel_name*
(Required) Specifies the node that you want to add to the cluster, by its name, that is displayed on the display panel.

Note: You cannot use the **-panelname** parameter with the **-wwnodename** parameter. However, you must use one of these parameters with the **addnode** command.

-wwnodename *wwnn_arg*
(Required) Specifies the node that you want to add to the cluster by the worldwide node name (WWNN) of the node.

Note: You cannot use the **-wwnodename** parameter with the **-panelname** parameter. However, you must use one of these parameters with the **addnode** command.

-name *new_name_arg*
(Optional) Specifies a name for the designated node.

-iogrp *iogroup_name* | *iogroup_id*
(Required) Specifies the I/O group that you want to add this node to.

Description

This command adds a new node to the cluster. You can obtain a list of candidate nodes (those that are not already assigned to a cluster) by issuing the **svcinfo lsnodecandidate** command.

The process of adding a node completes asynchronously. This means that while the node is in the adding state, the WWPN is not known and displays as zeros.

If the compatibility check fails, the following message displays:

CMMVC6201E The node could not be added, because incompatible software: status code [%1].

Prerequisites: Before you can add a node to the cluster, check the following requirements:

- The cluster must have more than one I/O group.
- The node that is being added to the cluster must use physical node hardware that has previously been used as a node in the cluster.
- The node that is being added to the cluster must use physical node hardware that has previously been used as a node in another cluster and both clusters are recognized by the same hosts.

  Attention: If the previous conditions apply, failure to follow the documented procedures can result in the corruption of all data that is managed by the cluster.

Adding a node: If you are adding the node into the cluster for the first time, you must record the node serial number, the WWNN, all WWPNs, and the I/O group to which it has been added. This can prevent possible data corruption if the node must be removed from and re-added to the cluster.

When a node is added to the cluster using the **svctask addnode** command or the cluster GUI, confirm that the node has previously been a member of the cluster. If it has, follow one of these two procedures:

- The node must be added back to the same I/O group that it was previously in. The WWNN of the nodes in the cluster can be determined using the **svcinfo lsnode** command. Or,
- If the WWPN of the node is not available, call the support team to add the node back into the cluster.

Optionally, you can assign a name to the new node. You can use this name in subsequent commands to refer to the node, instead of using the node ID. If you assign a label, this label is displayed as the node name from then on. If you do not assign a label, the default label is `nodeX`, where *X* is the node ID.

Applications on a host system direct I/O operations to file systems or logical volumes that are mapped by the operating system to vpaths. Vpaths are pseudo disk objects that are supported by the SDD; see the *Multipath Subsystem Device Driver: User's Guide* for more information.

The SDD maintains an association between a vpath and a VDisk. This association uses an identifier (UID), which is unique to the VDisk and is not reused. This enables the SDD to unambiguously associate vpaths with VDisks.

The SDD operates within a protocol stack, which also contains disk and fibre-channel device drivers that enable it to communicate with the cluster using the SCSI protocol over fibre-channel as defined by the ANSI FCS standard. The addressing scheme that is provided by these SCSI and fibre-channel device drivers uses a combination of a SCSI logical unit number (LUN) and the worldwide name for the fibre-channel node and ports.

In the event that errors occur, error recovery procedures (ERPs) operate at various tiers in the protocol stack. Some of these ERPs cause I/O to be redriven using the same WWNN and LUN numbers that were previously used.

The SDD does not check the association of the VDisk with the vpath on every I/O operation that it performs.

An invocation example

```
svcservicetask addnode -wwnodename 210000e08b053564 -iogrp io_grp0
```

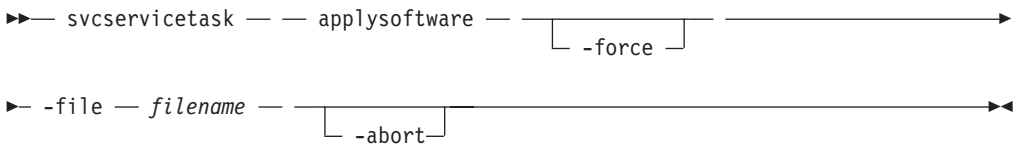
The resulting output

```
Node, id [6], successfully added
```

applysoftware

The **applysoftware** command upgrades the cluster to a new level of software.

Syntax



Parameters

-force
(Optional) Specifies that the upgrade continue even if there is a node in the I/O group that is not paired. The upgrade process forces the first node in each I/O group to shut down and upgrade.

Note: If the first node in the I/O group is not paired, the cluster becomes degraded and the data is lost even with the use of the **-force** parameter.

-file *filename*
(Required) Specifies that if you are doing an upgrade, you must designate the file name of the new software package.

-abort
(Optional) Stops the upgrade and backs out to where the application was before you started the upgrade.

Note: The **-force** parameter can be used with the **-abort** parameter. If one or more nodes are offline, you must use the **-force** parameter with the **-abort** parameter.

Description

This command starts the upgrade process of the cluster to a new level of software and applies to **svcservicetask** and **svcservicemodetask** commands. The **applysoftware** command applies a level of software to the node in both service and nonservice modes. In service mode, the **applysoftware** command is applied to the specific node. In nonservice mode, the **applysoftware** command is applied to the entire cluster.

The software package as specified by the file name must first be copied on to the current configuration node in the /home/admin/upgrade directory. You can use the PuTTy secure copy (scp) application to copy the file. See “PuTTY scp” for detailed information on this procedure.

The actual upgrade completes asynchronously.

The **svcinfo lssoftwaredumps** command allows you to view the contents of the /home/admin/upgrade directory.

The new package is moved internally from the **/home/admin/upgrade** directory and a checksum operation is processed on it. If the package fails the checksum operation, it is deleted and the upgrade fails. Otherwise, the package is extracted from the directory and the software upgrade begins.

An invocation example

```
svcservicetask applysoftware –file softwareupdate
```

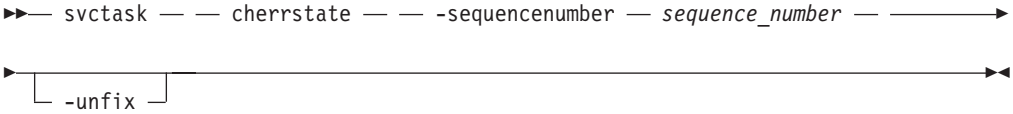
The resulting output

```
No feedback
```

cherrstate

The **cherrstate** command marks an unfixed error as fixed. You can also use it to mark a fixed error as unfixed.

Syntax



Parameters

-sequencenumber *sequence_number*
(Required) Specifies the error log sequence numbers to mark as fixed or as unfixed.

-unfix
(Optional) Specifies that the sequence numbers be marked as unfixed. Use this parameter when you have marked the wrong sequence number as fixed.

Description

The error log entries that the sequence numbers that you entered are marked as fixed. Use this command as a manual confirmation step that you have performed a maintenance procedure on the cluster, the fabric, or the subsystems.

This step is performed as part of the directed maintenance procedures (DMPs).

Optionally, if you have wrongly marked a sequence number as fixed, you can use the **-unfix** parameter to change the entry to unfixed.

An invocation example

```
svctask cherrstate -sequencenumber 2019
```

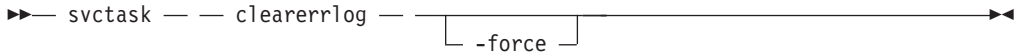
The resulting output

```
No feedback
```

clearerrlog

The **clearerrlog** command clears all entries from the error log including status events and any unfixed errors.

Syntax



Parameters

-force
(Optional) Specifies that the **clearerrlog** command be processed without confirmation requests. If the **-force** parameter is not supplied, you are prompted to confirm that you want to clear the log.

Description

This command clears all entries from the error log. The entries are cleared even if there are unfixed errors in the log. It also clears any status events that are in the log.

Attention: This command is destructive. Use it only use when you have either rebuilt the cluster or have fixed a major problem that has caused entries in the error log that you do not want to manually fix.

An invocation example

```
svctask clearerrlog -force
```

The resulting output

```
No feedback
```

dumperrlog

The **dumperrlog** command dumps the contents of the error log to a text file.

Syntax

Parameters

-prefix *filename_prefix*
(Optional) A file name is created from the prefix and a time stamp, and has the following format:

*prefix*_NNNNNN_YYMMDD_HHMMSS

where *NNNNNN* is the node front panel name.

Note: If the **-prefix** parameter is not supplied, the dump is directed to a file with a system-defined prefix of **errlog**.

Description

When run with no parameters, this command dumps the cluster error log to a file using a system-supplied prefix of **errlog**, which includes the node ID and time stamp. When a file name prefix is provided, the same operation is performed but the details are stored in the dumps directory within a file with a name that starts with the specified prefix.

A maximum of ten error-log dump files are kept on the cluster. When the 11th dump is made, the oldest existing dump file is overwritten.

Error log dump files are written to **/dumps/elogs**. The contents of this directory can be viewed using the **svcinfo lserrlogdumps** command.

Files are not deleted from other nodes until you issue the **cleardumps** command.

An invocation example

```
svcservicetask dumperrlog -prefix testerrorlog
```

The resulting output

```
No feedback
```

finderr

The **finderr** command analyzes the error log for the highest severity unfixed error.

Syntax

Description

The command scans the error log for any unfixed errors. Given a priority ordering within the code, the highest priority unfixed error is returned to standard output.

You can use this command to determine the order in which to fix the logged errors.

The Web-based directed maintenance procedures (DMPs) also use this command.

An invocation example

```
svctask finderr
```

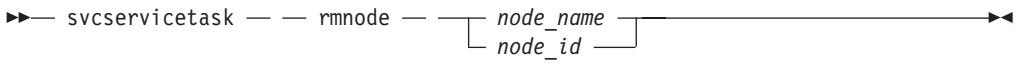
The resulting output

```
Highest priority unfixed error code is [1010]
```

rmnode

The **rmnode** command deletes a node from the cluster. You can enter this command any time after a cluster has been created.

Syntax



Parameters

node_name | node_id
Specifies the node to be deleted. The value for this parameter can be one of the following:

- The node name, that is, the label that you assigned when you added the node to the cluster
- The node ID that is assigned to the node [not the worldwide node name (WWNN)].

Description

This command removes a node from the cluster. This makes the node a candidate to be added back into this cluster or into another cluster. After the node is deleted, the other node in the I/O group enters write-through mode until another node is added back into the I/O group.

By default, the **rmnode** command flushes the cache on the specified node before the node is taken offline. In some circumstances, such as when the system is already degraded (for example, when both nodes in the I/O group are online and the virtual disks within the I/O group are degraded), the system ensures that data loss does not occur as a result of deleting the only node with the cache data.

The cache is flushed before the node is deleted to prevent data loss if a failure occurs on the other node in the I/O group.

Prerequisities:

Before you issue the **rmnode** command, perform the following tasks and read the following Attention notices to avoid losing access to data:

1. Determine which virtual disks (VDisks) are still assigned to this I/O group by issuing the following command. The command requests a filtered view of the VDisks, where the filter attribute is the I/O group.

   ```
   svcinfo lsvdisk -filtervalue IO_group_name=name
   ```

   where *name* is the name of the I/O group.

   Note: Any VDisks that are assigned to the I/O group that this node belongs to are assigned to the other node in the I/O group; the preferred node is changed. You cannot change this setting back.
2. Determine the hosts that the VDisks are mapped to by issuing the **svcinfo lsvdiskhostmap** command.
3. Determine if any of the VDisks that are assigned to this I/O group contain data that you need to access:
   - If you *do not* want to maintain access to these VDisks, go to step 5 on page 70.
   - If you *do* want to maintain access to some or all of the VDisks, back up the data or migrate the data to a different (online) I/O group.

4. Determine if you need to turn the power off to the node:
   - If this is the last node in the cluster, you do not need to turn the power off to the node. Go to step 5.
   - If this is *not* the last node in the cluster, turn the power off to the node that you intend to remove. This step ensures that the Subsystem Device Driver (SDD) does not rediscover the paths that are manually removed before you issue the delete node request.
5. Update the SDD configuration for each virtual path (vpath) that is presented by the VDisks that you intend to remove. Updating the SDD configuration removes the vpaths from the VDisks. Failure to update the configuration can result in data corruption. See the *Multipath Subsystem Device Driver: User's Guide* for details about how to dynamically reconfigure SDD for the given host operating system.
6. Quiesce all I/O operations that are destined for the node that you are deleting. Failure to quiesce the operations can result in failed I/O operations being reported to your host operating systems.

Attention:

1. Removing the last node in the cluster destroys the cluster. Before you delete the last node in the cluster, ensure that you want to destroy the cluster.
2. If you are removing a single node and the remaining node in the I/O group is online, the data can be exposed to a single point of failure if the remaining node fails.

Notes:

1. If you are removing the configuration node, the **rmnode** command causes the configuration node to move to a different node within the cluster. This process might take a short time, typically less than a minute. The cluster IP address remains unchanged, but any SSH client attached to the configuration node may need to reestablish a connection. The SAN Volume Controller Console reattaches to the new configuration node transparently.
2. If this is the last node in the cluster or if it is currently assigned as the configuration node, all connections to the cluster are lost. The user interface and any open CLI sessions are lost if the last node in the cluster is deleted. A time-out might occur if a command cannot be completed before the node is deleted.
3. This command may take some time to complete since the cache in the I/O group for that node is flushed before the node is removed.

An invocation example

```
svcservicetask rmnode 1
```

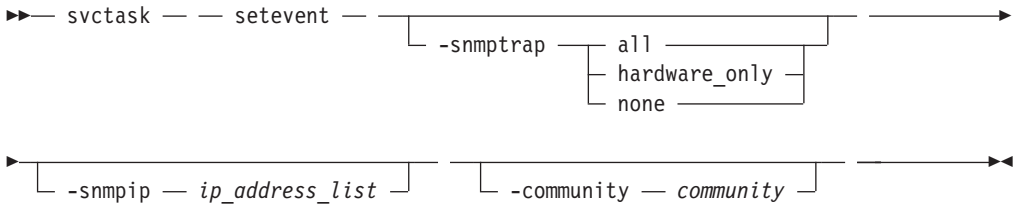
The resulting output

```
No feedback
```

setevent

The **setevent** command specifies how the SAN Volume Controller sends SNMP traps when an error or event is logged to the error log.

Syntax



Parameters

-snmptrap all | hardware_only | none
(Optional) Specifies the SNMP trap setting, which specifies when to receive a message that reports a problem or significant event.. You can set the following values for this parameter:

all Sends an SNMP trap for all errors and state changes that are logged.

hardware_only
Sends an SNMP trap for all errors, but not for object state changes.

none Does not send any SNMP traps or errors. This is the default setting for a new cluster.

-snmpip *ip_address_list*
(Optional) Specifies the IP addresses of host systems running the SNMP manager software. You can specify up to six IP addresses, using one of following formats:

- Colon-separated list of IPv4 addresses
- Comma-separated list of IPv6 addresses
- Comma-separated list of IPv4 or IPv6 addresses, including an optional port number for each address. For example:
  - For IPv4: 9.20.83.124:80,9.20.83.125:81
  - For IPv6: [FFE0:1234::1]:80,[FFE0:1234::2]:81

Entries in excess of the number specified using the **-community** parameter are ignored.

-community *community*
(Optional) Specifies the SNMP community string. This is a colon-separated list of values with up to six items per list. The maximum length of the community string that is used in SNMP trap generation cannot be more than 60 characters.

Description

This command sets or modifies error log settings, which define the actions to take when errors and events are logged.

You can use this command to setup SNMP traps. For SNMP, you must supply the following information:

- When to raise a trap.
- The IP address of the SNMP manager
- The SNMP community

Table 3 on page 72 provides IPv4 and IPv6 formats that are supported for the **-snmpip** *ip_address_list* parameter:

Table 3. SNMP ip_address_list formats

IP type	*ip_address_list* **format**
IPv4 (no port set, SAN Volume Controller uses default port of 162)	`1.2.3.4`
IPv4 with specific port	`1.2.3.4:22`
Full IPv6 (no port set, SAN Volume Controller uses default port of 162)	`2002:914:fc12:848:209:6bff:fe8c:4ff6`
Full IPv6 with port	`[2002:914:fc12:848:209:6bff:fe8c:4ff6]:23`
Zero-compressed IPv6 (no port set, SAN Volume Controller uses default port of 162)	`2002::4ff6`
Zero-compressed IPv6 with port	`[2002::4ff6]:23`

An invocation example

```
svctask setevent -snmptrap all -snmpip 1.2.3.4
-community mysancommunity
```

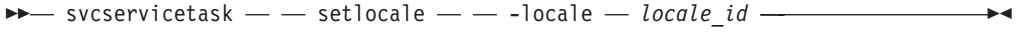
The resulting output

```
No feedback
```

setlocale

The **setlocale** command changes the locale setting for the cluster. It also changes command output to the chosen language.

Syntax



Parameters

-locale *locale_id*
Specifies the locale ID.

Description

This command changes the language in which error messages are displayed as output from the command-line interface. Subsequently, all error messages from the command-line tools are generated in the chosen language. This command is run when you request a change of language (locale) and is generally run from the Web page. Issue the **svcservicetask setlocale** command to change the locale setting for the cluster; all interface output is changed to the chosen language. For example, to change the language to Japanese, type the following:

```
svcservicetask setlocale -locale 3
```

where **3** is the value for Japanese. The following values are supported:

- **0** US English (default)
- **3** Japanese

Note: This command does not change the front panel display panel settings.

An invocation example

```
svcservicetask setlocale -locale 3
```

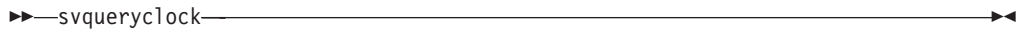
The resulting output

```
No feedback
```

svqueryclock

The **svqueryclock** command returns the date, time, and current time-zone of the cluster.

Syntax



Description

This command returns the date, time and current time-zone of the cluster.

An invocation example

```
svqueryclock
```

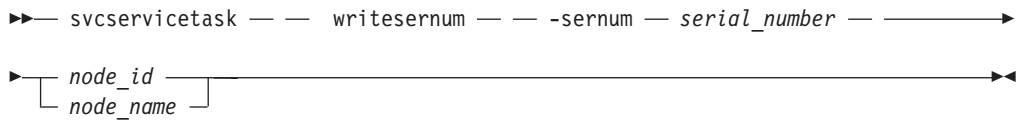
The resulting output

```
Mon Nov 25 14:59:28 GMT 2002
```

writesernum

Use the **writesernum** command to write the node serial number into the planar NVRAM.

Syntax



Parameters

-sernum *serial_number*
(Required) Specifies the serial number to write to the nonvolatile memory of the system planar.

node_id | *node_name*
(Required) Specifies the node where the system planar is located. The serial number is written to this system planar. This name is not the worldwide node name (WWNN).

Description

This command writes the node serial number into the planar NVRAM and then reboots the system. You can find the serial number at the front of the node without

having to remove it from the rack. The serial number is located to the left of the right-side thumbscrew that holds the node into the rack. This serial number is usually seven digits.

Note: Once you have written the serial number to the planar NVRAM, you can issue the **svcinfo lsnodevpd** command to verify that the number is correct. The `system_serial_number` field contains the serial number.

An invocation example

```
svcservicetask writesernum -sernum 1300027 node1
```

The resulting output

```
No feedback
```

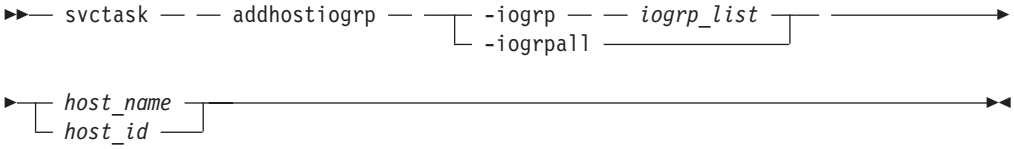
Chapter 10. Host commands

The following commands enable you to work with host options with the SAN Volume Controller.

addhostiogrp

The **addhostiogrp** command enables you to map I/O groups to an existing host object.

Syntax



Parameters

-iogrp *iogrp_list*
(Required if you do not use **-iogrpall**) Specifies a colon-separated list of one or more I/O groups that must be mapped to the host. You cannot use this parameter with the **-iogrpall** parameter.

-iogrpall
(Required if you do not use **-iogrp**) Specifies that all the I/O groups must be mapped to the specified host. You cannot use this parameter with the **-iogrp** parameter.

host_id | host_name
(Required) Specifies the host to which the I/O groups must be mapped, either by ID or by name.

Description

This command allows you to map the list of I/O groups to the specified host object.

An invocation example

```
svctask addhostiogrp -iogrpall testhost
```

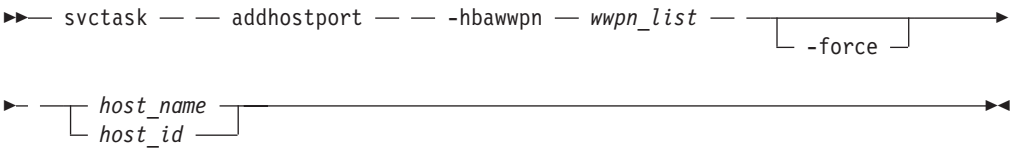
The resulting output

```
No feedback
```

addhostport

The **addhostport** command adds worldwide port names (WWPNs) to an existing host object.

Syntax



Parameters

-hbawwpn *wwpn_list*
(Required) Specifies the list of ports to add to the host.

-force
(Optional) Specifies that the list of ports be added to the host without the validation of any WWPNs.

host_id | *host_name*
(Required) Specifies the host object to add ports to, either by ID or by name.

Description

This command adds the list of HBA WWPNs to the specified host object. Only logged-in unconfigured WWPNs can be added. For a list of candidate WWPNs, see the **svcinfo lshbaportcandidate** command.

Some HBA device drivers do not log in to the fabric until they can recognize target LUNs. Because they do not log in, their WWPNs are not be recognized as candidate ports. You can specify the **-force** parameter with the **svctask addhostport** command to stop the validation of the WWPN list.

Any virtual disks that are mapped to this host object automatically map to the new ports.

Replacing an HBA in a host: List the candidate HBA ports by issuing the **svcinfo lshbaportcandidate** command, This command generates a list of the HBA ports that are available for addition to host objects. One or more of these ports should correspond with the one or more WWPNs that belong to the new HBA. Locate the host object that corresponds with the host in which you have replaced the HBA. The following command lists all the defined host objects:

```
svcinfo lshost
```

To list the WWPNs that are currently assigned to the host, issue the following command:

```
svcinfo lshost hostobjectname
```

where *hostobjectname* is the name of the host object.

Add the new ports to the existing host object by issuing the following command:

```
svctask addhostport -hbawwpn one or more existing WWPNs
separated by : hostobjectname/ID
```

where *one or more existing WWPNs separated by :* and *hostobjectname/id* correspond to those values that are listed in the previous steps.

Remove the old ports from the host object by issuing the following command:

```
svctask rmhostport -hbawwpn one or more existing WWPNs
separated by : hostobjectname/ID
```

where *one or more existing WWPNs separated by :* corresponds to those values that are listed in the previous step that belong to the old HBA that has been replaced. Any mappings that exist between the host object and VDisks are automatically applied to the new WWPNs. Therefore, the host can recognize that the VDisks as the same SCSI LUNs as before. See the host multipathing device driver documentation for additional information about dynamic reconfiguration.

An invocation example

```
svctask addhostport -hbawwpn 210100E08B251DD4 host_one
```

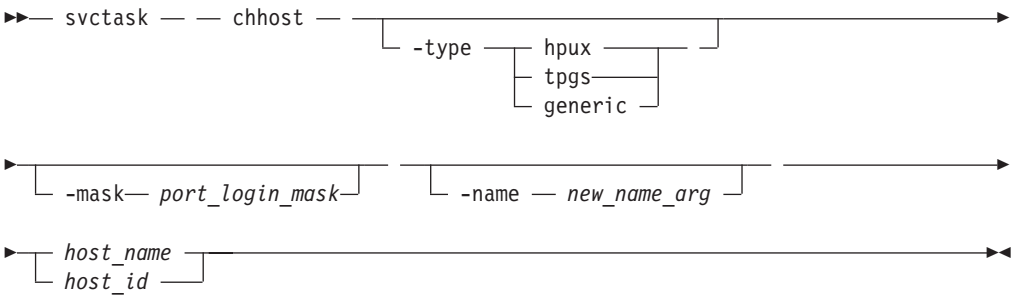
The resulting output

```
No feedback
```

chhost

The **chhost** command changes the name or type of a host object. This does not affect any existing virtual disk-to-host mappings.

Syntax



Parameters

-type hpux | tpgs | generic
(Optional) Specifies the type of host: **hpux**, **tpgs**, or **generic**. The default is **generic**. The **tpgs** parameter enables extra target port unit attentions. See the *IBM System Storage SAN Volume Controller Host Attachment Guide* for more information on the hosts that require the **-type** parameter.

-name *new_name_arg*
(Optional) Specifies the new name that you want to assign to the host object.

-mask *port_login_mask*
(Optional) Specifies which node target ports that a host can access. The port mask is four binary bits and is made of a combination of 0's and 1's, where 0 indicates that the corresponding target port cannot be used and 1 indicates that it can be used. The right-most bit in the mask corresponds to the lowest numbered target port (1 not 4) on a node. Valid mask values range from **0000** (no ports enabled) to **1111** (all ports enabled). For example, a mask of **0011** enables port 1 and port 2. The default value is **1111** (all ports enabled).

host_name | host_id
(Required) Specifies the host object to modify, either by ID or by current name.

Description

This command can change the name of the specified host to a new name, or it can change the type of host. This command does not affect any of the current virtual disk-to-host mappings.

The port mask applies to logins from the host initiator port that are associated with the host object. For each login between a host HBA port and node port, the node examines the port mask that is associated with the host object for which the host HBA is a member and determines if access is allowed or denied. If access is denied, the node responds to SCSI commands as if the HBA port is unknown.

An invocation example

```
svctask chhost -name testhostlode -mask 0011 hostone
```

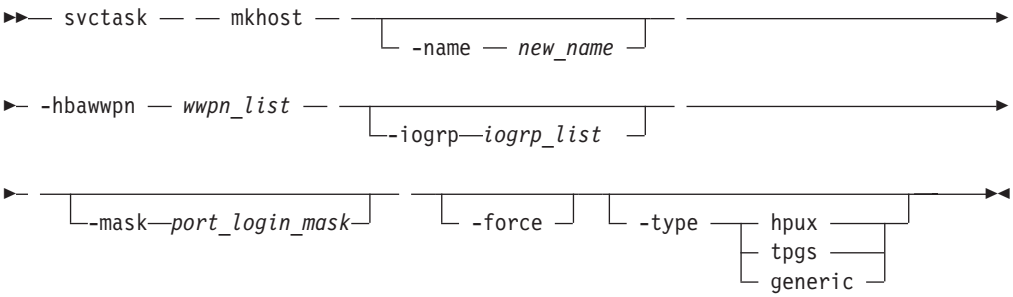
The resulting output

```
No feedback
```

mkhost

The **mkhost** command creates a logical host object.

Syntax



Parameters

-name *new_name*
(Optional) Specifies a name or label for the new host object.

-hbawwpn *wwpn_list*
(Required) Specifies a list of host bus adapter (HBA) worldwide port names (WWPNs) to add to the specified host object.

-iogrp *iogrp_list*
(Optional) Specifies a set of one or more I/O groups that the host can access the VDisks from. I/O groups are specified using their names or IDs, separated by a colon. Names and IDs can be mixed in the list. If this parameter is not specified, the host is associated with all I/O groups.

-mask *port_login_mask*
(Optional) Specifies which node target ports that a host can access. The port mask is four binary bits and is made up of a combination of 0's and 1's, where 0 indicates that the corresponding target port cannot be used and 1 indicates

that it can be used.. The right-most bit in the mask corresponds to the lowest numbered target port (1 not 4) on a node. Valid mask values range from **0000** (no ports enabled) to **1111** (all ports enabled). For example, a mask of **0011** enables port 1 and port 2. The default value is **1111** (all ports enabled).

-force
(Optional) Specifies that a logical host object be created without validation of the WWPNs.

-type hpux | tpgs | generic
(Optional) Specifies the type of host: **hpux**, **tpgs**, or **generic**. The default is **generic**. The **tpgs** parameter enables extra target port unit attentions. See the *IBM System Storage SAN Volume Controller Host Attachment Guide* for more information on the hosts that require the **-type** parameter.

Description

The **mkhost** command associates one or more HBA WWPNs with a logical host object. This command creates a new host. The ID is displayed when the command completes. You can subsequently use this object when you map virtual disks to hosts by using the **mkvdiskhostmap** command.

Issue the **mkhost** command only once. The cluster scans the fabric for WWPNs in the host zone. The cluster itself cannot filter into the hosts to determine which WWPNs are in which hosts. Therefore, you must use the **svctask mkhost** command to identify the hosts.

After you identify the hosts, mappings are created between hosts and virtual disks. These mappings effectively present the virtual disks to the hosts to which they are mapped. All WWPNs in the host object are mapped to the virtual disks.

Some HBA device drivers do not log in to the fabric until they can see target logical unit numbers (LUNs). Because they do not log in, their WWPNs are not recognized as candidate ports. You can specify the **-force** parameter with this command to stop the validation of the WWPN list.

This command fails if you add the host to an I/O group that is associated with more host ports or host objects than is allowed by the limits within the cluster.

For additional information, see the **svctask mkvdiskhostmap** and **svcinfo lshbaportcandidate** commands.

An invocation example

```
svctask mkhost -name hostone -hbawwpn 210100E08B251DD4 -force -mask 1001
```

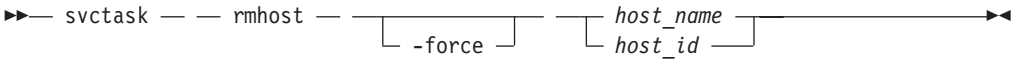
The resulting output

```
Host id [1] successfully created.
```

rmhost

The **rmhost** command deletes a host object.

Syntax



Parameters

-force
(Optional) Specifies that you want the system to delete the host object even if mappings still exist between this host and virtual disks (VDisks). When the **-force** parameter is specified, the mappings are deleted before the host object is deleted.

host_name | host_id
(Required) Specifies the host object to delete, either by ID or by name.

Description

The **rmhost** command deletes the logical host object. The WWPNs that were contained by this host object (if it is still connected and logged in to the fabric) are returned to the unconfigured state. When you issue the **svcinfo lshbaportcandidate** command, the host objects are listed as candidate ports.

If any mappings still exist between this host and virtual disks, the command fails unless you specify the **-force** parameter. When the **-force** parameter is specified, the **rmhost** command deletes the mappings before the host object is deleted.

An invocation example

```
svctask rmhost host_one
```

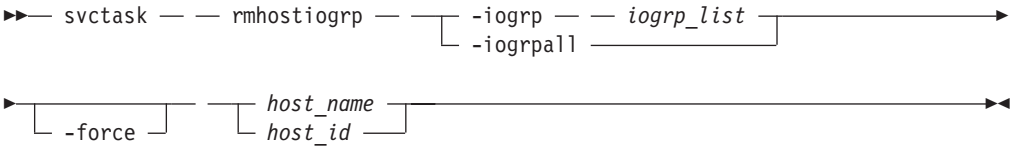
The resulting output

```
No feedback
```

rmhostiogrp

The **rmhostiogrp** command enables you to delete mappings between one or more I/O groups and a specified host object.

Syntax



Parameters

-iogrp *iogrp_list*
(Required) Specifies a set of one or more I/O group mappings that will be deleted from the host. You cannot use this parameter with the **-iogrpall** parameter.

-iogrpall
(Optional) Specifies that all the I/O group mappings that are associated with the specified host must be deleted from the host. You cannot use this parameter with the **-iogrp** parameter.

-force
(Optional) Specifies that you want the system to remove the specified I/O group mappings on the host even if the removal of a host to I/O group mapping results in the loss of VDisk-to-host mappings.

host_id | host_name
(Required) Specifies the identity of the host either by ID or name from which the I/O group mappings must be deleted.

Description

The **rmhostiogrp** command deletes the mappings between the list of I/O groups and the specified host object.

This command fails if any of the I/O groups that are deleted from the host contain VDisks that have host mappings to the host. To resolve this problem, do one of the following:

- Delete the VDisks-to-host mappings that are causing the error.
- Delete the VDisks or the host.
- Issue the **rmhostiogrp** command with the **-force** parameter.

An invocation example

```
svctask rmhostiogrp -iogrp 1:2 host0
```

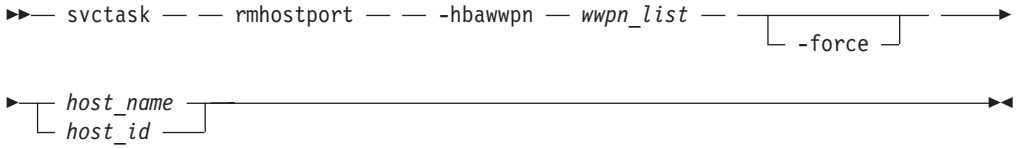
The resulting output

```
No feedback
```

rmhostport

The **rmhostport** command deletes worldwide port names (WWPNs) from an existing host object.

Syntax



Parameters

-hbawwpn *wwpn_list*
(Required) Specifies the list of ports that you can delete from the host.

-force
(Optional) Specifies that you want the system to delete the ports that you have specified without performing the validation check. The validation check ensures that the list of ports that you want to delete are actually mapped to the specified host. When the ports are deleted, they become unconfigured WWPNs.

host_name | host_id
(Required) Specifies the host name or the host ID.

Description

This command deletes the list of HBA WWPNs from the specified host object. If these ports are still logged in to the fabric, they become unconfigured and are listed as candidate WWPNs. See also the **svcinfo lshbaportcandidate** command.

Any virtual disks that are mapped to this host object are automatically unmapped from the ports.

Replacing an HBA in a host: List the candidate HBA ports by issuing the **svcinfo lshbaportcandidate** command. A list of the HBA ports that are available to be added to host objects is displayed. One or more of these ports corresponds with one or more WWPNs that belong to the new HBA. Locate the host object that corresponds to the host in which you have replaced the HBA. The following command lists all the defined host objects:

```
svcinfo lshost
```

To list the WWPNs that are currently assigned to the host, issue the following:

```
svcinfo lshost hostobjectname
```

where *hostobjectname* is the name of the host object.

Add the new ports to the existing host object by issuing the following command:

```
svctask addhostport -hbawwpn one or more existing WWPNs
separated by : hostobjectname/ID
```

where *one or more existing WWPNs separated by :* and *hostobjectname/id* correspond to those values listed in the previous steps.

Remove the old ports from the host object by issuing the following command:

```
svctask rmhostport -hbawwpn one or more existing WWPNs
separated by : hostobjectname/ID
```

where *one or more existing WWPNs separated by :* corresponds with those WWPNs that are listed in the previous step that belong to the old HBA that has been replaced. Any mappings that exist between the host object and VDisks are automatically applied to the new WWPNs. Therefore, the host recognizes that the VDisks are the same SCSI LUNs as before. See the *Multipath Subsystem Device Driver: User's Guide* for additional information about dynamic reconfiguration.

An invocation example

```
svctask rmhostport -hbawwpn 210100E08B251DD4 host_one
```

The resulting output

```
No feedback
```

Chapter 11. Virtual disk commands

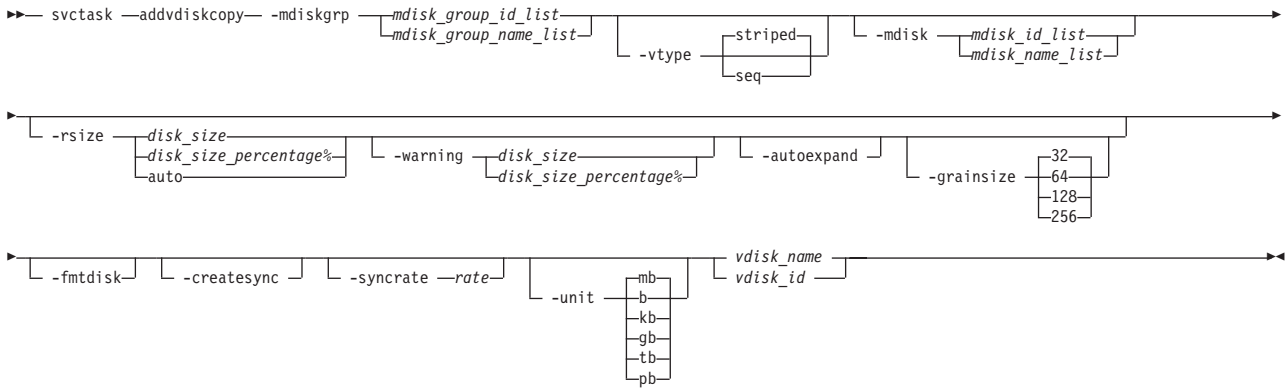
The following commands enable you to work with virtual disk options with the SAN Volume Controller.

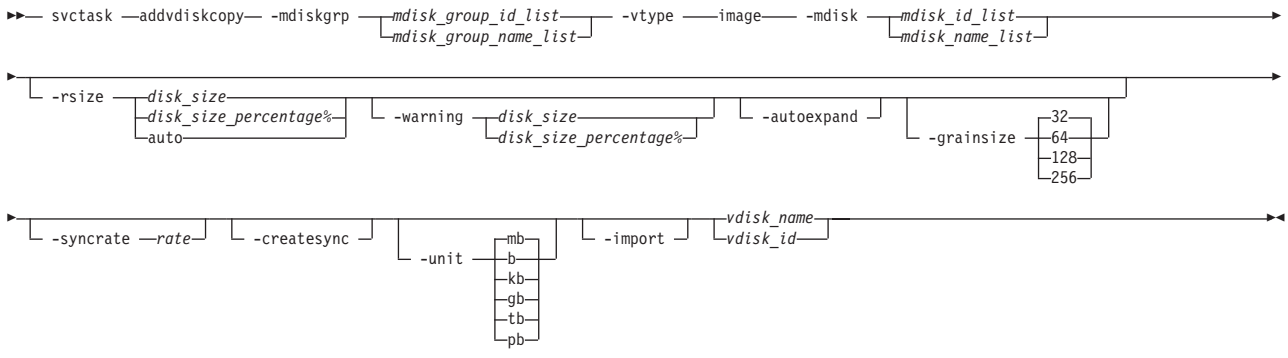
addvdiskcopy

The **addvdiskcopy** command adds a copy to an existing VDisk, which changes a nonmirrored VDisk into a mirrored VDisk.

Note: The first syntax diagram depicts the addition of a sequential or striped mode virtual disk. The second syntax diagram depicts the addition of an image mode virtual disk.

Syntax





Parameters

-mdiskgrp *mdisk_group_id_list* | *mdisk_group_name_list*
(Required) Specifies the managed disk groups to use to create copies for the virtual disk. You must specify a group for each copy that is being added.

-vtype seq | striped | image
(Optional) Specifies the virtualization type for the copy: sequential, striped, or image. The type can be different than the virtualization types for other copies on the VDisk. The default virtualization type is **striped**.

-mdisk *mdisk_id_list | mdisk_name_list*
(Optional) Specifies one or more managed disks (MDisks). For sequential and image mode copies, you must specify a single MDisk that has sufficient free extents. For image mode copies, the MDisk must be in unmanaged mode. For sequential mode copies the MDisk must be in the managed mode.

-syncrate *rate*
(Optional) Specifies the copy synchronization rate. A value of zero (**0**) prevents synchronization. The default value is **50**. For the supported **-syncrate** values and their corresponding rates, see Table 4 on page 86.

-createsync
(Optional) Suppresses the synchronization of the new VDisk copy with the primary copy. Using this parameter can cause data corruption if the primary copy fails and leaves an unsynchronized secondary copy to provide data. Using this parameter can cause loss of read stability in unwritten areas if the primary copy fails, data is read from the primary copy, and then different data is read from the secondary copy. To avoid data loss or read stability loss, use this parameter only for a primary copy that has been formatted and not written to, and with the **-fmtdisk** parameter.

-fmtdisk
(Optional) Formats a sequential or striped mode copy. You must also specify the **-createsync** parameter, which labels the formatted copy as identical to the primary copy. The **-fmtdisk** parameter causes the VDisk to go offline until new VDisk copy formatting completes. To query the formatting progress, use the **lsvdiskprogress** command.

-rsize *disk_size | disk_size_percentage***% | auto**
(Optional) Makes the copy space-efficient and specifies the real size of the copy. Specify the *disk_size | disk_size_percentage* value using an integer, or an integer immediately followed by the percent character (**%**). The default units for *disk_size* are megabytes (MB); to specify different units, use the **-unit** parameter. The **auto** option creates a VDisk copy that uses the entire size of the MDisk; if you specify the **-rsize auto** option, you must also specify the **-vtype image** option.

-warning *disk_size | disk_size_percentage***%**
(Optional) Requires that the **-rsize** parameter also be specified. Generates a warning when the used disk capacity on the space-efficient copy first exceeds the specified threshold. You can specify a *disk_size* integer, which defaults to megabytes (MB) unless the **-unit** parameter is specified; or you can specify a *disk_size***%**, which is a percentage of the virtual disk size. If **-autoexpand** is enabled, the default value for **-warning** is 80% of the virtual disk capacity. If **-autoexpand** is not enabled, the default value for warning is 80% of the real capacity. To disable warnings, specify **0**.

-autoexpand
(Optional) Requires that the **-rsize** parameter also be specified. Specifies that space-efficient copies automatically expand their real capacities by allocating new extents from their managed disk group. If the **-autoexpand** parameter is specified, the **-rsize** parameter specifies a capacity that is reserved by the copy. This protects the copy from going offline when its managed disk group runs out of space by allowing it to consume this reserved space first.

-grainsize 32 | 64 | 128 | 256
(Optional) Requires that the **-rsize** parameter also be specified. Sets the grain size (KB) for a space-efficient VDisk. The default is 32 KB.

-unit b | kb | mb | gb | tb | pb
(Optional) Specifies the data units for the **-rsize** and **-warning** parameters.

-import
(Optional) Imports an image mode disk that contains a space-efficient volume into the cluster. Requires that the **-rsize** and **-vtype image** parameters also be specified.

vdisk_name | vdisk_id
(Required) Specifies the virtual disk to add the VDisk copy to, either by ID or by name.

Description

The **addvdiskcopy** command adds a copy to an existing VDisk, which changes a nonmirrored VDisk into a mirrored VDisk. Use the **-mdiskgrp** parameter to specify the managed disk group that will provide storage for the copy; the **svcinfo lsmdiskgrp** command lists the available managed disk groups and the amount of available storage in each group.

The virtualization types are defined as follows:

sequential (seq)
This policy requires the **-mdisk** parameter with a single managed disk as its argument. This MDisk must be in the managed mode.

It creates the virtual disk using extents from the given managed disk (assuming there are enough free extents on the managed disk).

striped
This is the default policy. If the **-vtype** parameter is not specified, this policy is used in its default form. That is, all managed disks in the managed disk group are used to create the virtual disk. The striping is at an extent level; one extent from each managed disk in the group is used. For example, a managed disk group with 10 managed disks uses one extent from each managed disk, then it uses the 11th extent from the first managed disk, and so on.

If the **-mdisk** parameter is also specified, you can supply a list of managed disks to use as the stripe set. This can be two or more managed disks from the same managed disk group. The same circular algorithm is used across the striped set. However, a single managed disk can be specified more than once in the list. For example, if you enter `-m 0:1:2:1`, the extents are from the following **managed** disks: 0, 1, 2, 1, 0, 1, 2, and so forth. All MDisks that are specified in the **-mdisk** parameter must be in **managed** mode.

image This policy allows image mode virtual disks to be created when a managed disk already has data on it, perhaps from a previrtualized subsystem. When an image mode virtual disk is created, it directly corresponds to the (previously unmanaged) managed disk that it was created from; therefore, virtual disk logical block address (LBA) *x* equals managed disk LBA *x*. You can use this command to bring a nonvirtualized disk under the control of the cluster. After it is under the control of the cluster, you can migrate the virtual disk from the single managed disk. When it is migrated, the virtual disk is no longer an image mode virtual disk.

You can add image mode VDisks to an already populated MDisk group with other types of VDisks, such as a striped or sequential.

Note: An image mode copy must be at least as large as the VDisk that it is being added to, but any capacity beyond the size of the VDisk is not accessible.

The command returns the ID of the newly created VDisk copy.

Table 4 provides the relationship of the *rate* value to the data copied per second.

Table 4. Relationship between the rate value and the data copied per second

User-specified *rate* **attribute value**	**Data copied/sec**
1 - 10	128 KB
11 - 20	256 KB
21 - 30	512 KB
31 - 40	1 MB
41 - 50	2 MB
51 - 60	4 MB
61 - 70	8 MB
71 - 80	16 MB
81 - 90	32 MB
91 - 100	64 MB

An invocation example

```
svctask addvdiskcopy -mdiskgrp 0 vdisk8
```

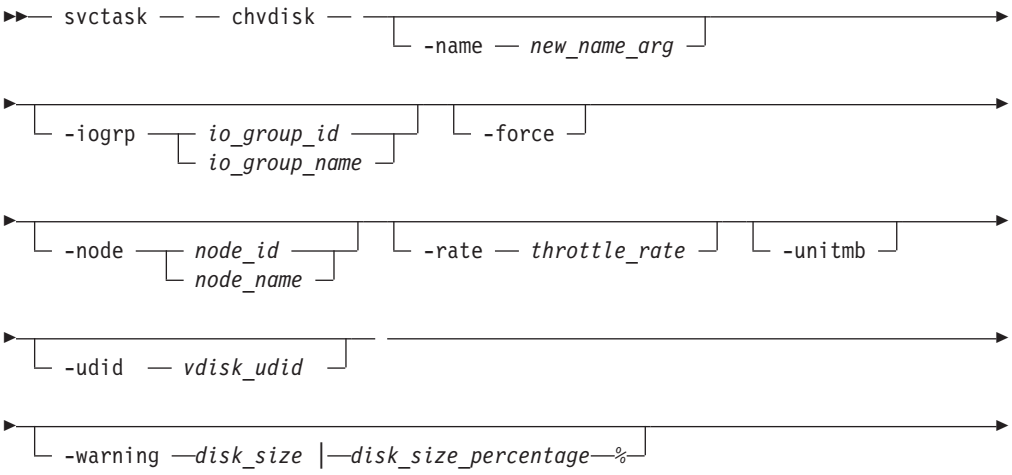
The resulting output

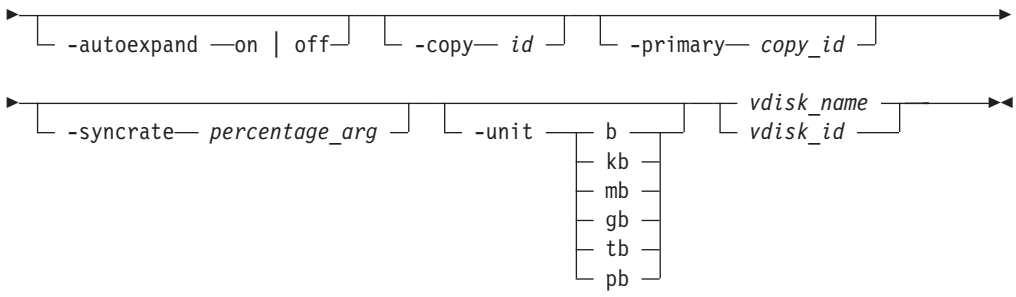
```
Vdisk [8] copy [1] successfully created
```

chvdisk

The **chvdisk** command modifies the properties of a virtual disk, such as the disk name, I/O group, I/O governing rate, or unit number.

Syntax





Parameters

-name *new_name_arg*

(Optional) Specifies a new name to assign to the virtual disk. You cannot use this parameter with the **-iogrp**, **-rate**, **-node**, or **-udid** parameters. This parameter is required if you do not use the **-iogrp**, **-rate**, or **-udid** parameter.

-iogrp *io_group_id | io_group_name*

(Optional) Specifies a new I/O group to move the virtual disk to, by IO group ID or IO group name. You can use the **-node** parameter with the **-iogrp** parameter to specify a preferred node for the specified VDisk.

Notes:

1. If the VDisk has a mapping to any hosts, it is not possible to move the VDisk to an I/O group that does not include any of those hosts.
2. This parameter can fail if there is not enough space to allocate bitmaps for a mirrored VDisk in the target IO group.
3. This parameter can fail if any copy is not synchronized. The **-force** parameter can be used to force the move, but this resynchronizes the VDisk.
4. If the VDisk is offline, use one of the **recovervdisk** commands to recover the VDisk and bring it back online. Beginning with SAN Volume Controller version 4.3.1, use of the recovery I/O group is not required.

-force

(Optional) Forces the VDisk to be removed from an I/O group. This parameter can only be used with the **-iogrp** parameter.

Attention:

1. If the **-force** parameter is used and the cluster is unable to destage all write data from the cache, the contents of the VDisk are corrupted by the loss of the cached data.
2. If the **-force** parameter is used to move a VDisk that has out-of-sync copies, a full resynchronization is required.

-rate *throttle_rate* **[-unitmb]**

(Optional) Specifies the I/O governing rate for the VDisk, which caps the amount of I/O that is accepted. The default *throttle_rate* units are I/Os. To change the *throttle_rate* units to megabytes per second (MBps), specify the **-unitmb** parameter. The governing rate for a virtual disk can be specified by I/Os or by MBps, but not both. However, you can set the rate to I/Os for some virtual disks and to MBps for others.

You cannot use this parameter with the **-name**, **-iogrp**, **-node**, or **-udid** parameters.

-udid *vdisk_udid*
(Optional) Specifies the unit number (**udid**) for the disk. The *vdisk_udid* is an identifier that is required to support OpenVMS hosts; no other systems use this parameter. Valid options are a decimal number from 0 to 32 767 or a hexadecimal number from 0 to 0x7FFF. A hexadecimal number must be preceded by **0x** (for example, **0x1234**). If you do not use the **-udid** parameter, the default **udid** is **0**.

You cannot use this parameter with the **-name**, **-iogrp**, **-node**, or **-rate** parameters.

-warning *disk_size* | *disk_size_percentage***%**
(Optional) Generates a warning when the used disk capacity on the space-efficient copy first exceeds the specified threshold. You can specify a *disk_size* integer, which defaults to MBs unless the **-unit** parameter is specified; or you can specify a *disk_size***%**, which is a percentage of the virtual disk size. To disable warnings, specify **0** or **0%**.

-unit b | kb | mb | gb | tb | pb
(Optional) Specifies the data units to use for the **-warning** *disk_size* parameter.

-autoexpand on | off
(Optional) Specifies whether space-efficient VDisk copies automatically expand their real capacities by allocating new extents from their managed disk group. To use this parameter, the VDisk must be space-efficient.

-copy *id*
(Optional) Specifies the copy to apply the changes to. You must specify this parameter with the **-autoexpand** or **-warning** parameter. The **-copy** parameter is required if the specified VDisk is mirrored and only one VDisk copy is space-efficient. If both copies are space-efficient and the **-copy** parameter is not specified, the specified **-autoexpand** or **-warning** parameter is set on both copies.

-primary *copy_id*
(Optional) Specifies the primary copy. Changing the primary copy only takes effect when the new primary copy is online and synchronized. If the new primary is online and synchronized when the command is issued, the change takes effect immediately.

-syncrate *percentage*
(Optional) Specifies the copy synchronization rate, as a percentage of the peak synchronization rate. A value of zero (**0**) prevents synchronization.

-node *node_id* | *node_name*
(Optional) Specifies a preferred node for the specified VDisk. When using this parameter, you must also specify the **-iogrp** parameter. You cannot use this parameter with the **-name**, **-rate**, or **-udid** parameters.

vdisk_name | *vdisk_id*
(Required) Specifies the virtual disk to modify, either by ID or by name.

Description

The **chvdisk** command modifies a single property of a virtual disk (VDisk). To change the VDisk name and modify the I/O group, for example, you must issue the command twice.

You can specify a new name or label. You can use the new name subsequently to refer to the virtual disk. To specify a preferred node for the VDisk, use the **-node** *node_id | node_name* parameter.

You can change the I/O group with which this virtual disk is associated. However, to change the I/O group, you must first flush the cache within the nodes in the current I/O group to ensure that all data is written to disk. Ensure that you suspend I/O operations at the host level before you perform this operation.

Attention:

1. Do not move a VDisk to an offline I/O group under any circumstance. To avoid any data loss, you must ensure that the I/O group is online before you move the VDisks.
2. Do not move an offline VDisk to the recovery I/O group. Beginning with SAN Volume Controller version 4.3.1, use of the recovery I/O group is not required. Instead, use one of the **recovervdisk** commands to recover the VDisk and bring it back online.

You can set a limit on the amount of I/O transactions that is accepted for this virtual disk. It is set in terms of I/Os per second or MBs per second. By default, no I/O governing rate is set when a virtual disk is created.

Attention: All capacities, including changes, must be in multiples of 512 bytes. An error occurs if you specify a capacity that is not a multiple of 512, which can only happen when byte units (**-b**) are used. The default capacity is in MB.

When the virtual disk is created, there is no throttling applied to it. Using the **-rate** parameter can change this. To change the virtual disk back to an unthrottled state, specify 0 (zero) with the **-rate** parameter.

You can migrate a VDisk to a new I/O group to manually balance the workload across the nodes in the cluster. You might end up with a pair of nodes that are overworked and another pair that are underworked. Use the following procedure to migrate a single VDisk to a new I/O group. Repeat for other VDisks as required.

Attention: This is a disruptive procedure. Access to the VDisk is lost while you follow this procedure.

Ensure that when you migrate a VDisk to a new I/O group, you quiesce all I/O operations for the VDisk. Determine the hosts that are using this VDisk. Stop and delete any FlashCopy mappings or Metro or Global Mirror relationships that use this VDisk. To check if the VDisk is part of a relationship or mapping, issue the **svcinfo lsvdisk** *vdiskname | id* command, where *vdiskname | id* is the name or ID of the VDisk.

Look for the **FC_id** and **RC_id** fields. If these are not blank, the VDisk is part of a mapping or relationship. See the FlashCopy commands or Metro Mirror and Global Mirror commands for details on how to stop or delete the mapping or relationship. Issue the following command to migrate the VDisk:

```
svctask chvdisk -iogrp newiogrpname|id vdiskname|id
```

Follow the procedure to discover the new vpaths and to check that each vpath is presenting the correct number of paths. See the *Multipath Subsystem Device Driver: User's Guide* for details on how to dynamically reconfigure SDD for the given host operating system.

Note: The command fails if you attempt to change the primary copy of a mirrored VDisk while the **repairvdiskcopy -resync** command is running.

An invocation example

```
svctask chvdisk -rate 2040 -unitmb 6
```

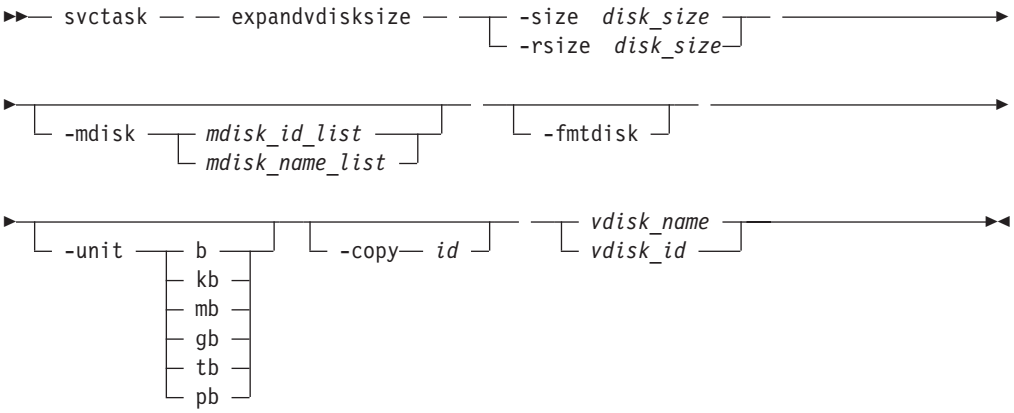
The resulting output

```
No feedback
```

expandvdisksize

The **expandvdisksize** command expands the size of a VDisk by a given capacity.

Syntax



Parameters

-size *disk_size*
(Optional) Specifies the capacity by which the virtual disk is expanded. Disk size is used with the value of the unit. All capacities, including changes must be in multiples of 512 bytes. An error occurs if you specify a capacity that is not a multiple of 512, which can only occur when byte units (**-unit b**) are used. However, an entire extent is reserved even if it is only partially used. The default *disk_size* unit is megabytes (MB). You cannot specify the **-size** parameter with the **-rsize** parameter. You must specify either **-size** or **-rsize**. If the VDisk is space-efficient, MDisks cannot be specified.

-rsize *disk_size*
(Optional) Specifies the capacity by which to increase the real size of a space-efficient VDisk. Specify the *disk_size* value using an integer. Specify the unit for a *disk_size* integer using the **-unit** parameter; the default unit is megabytes (MB). The **-rsize** value can be greater than, equal to, or less than the size of the VDisk. You cannot specify the **-rsize** parameter with the **-size** parameter. You must specify either **-size** or **-rsize**.

-copy *id*
(Optional) Specifies the copy to change the real capacity for. You must also

specify the **-rsize** parameter; you can only modify the real capacity of a VDisk copy. The **-copy** parameter is required if the specified VDisk is mirrored and only one copy is space-efficient. If the VDisk is mirrored, both copies are space-efficient and **-copy** is not specified, both copies are modified by the same amount.

-mdisk *mdisk_id_list | mdisk_name_list*
(Optional) Specifies the list of one or more MDisks to be used as the stripe set. The extents that expand the VDisk come from the specified list of MDisks. All MDisks in the list must be part of the same MDisk group. The **-mdisk** parameter cannot be used if the specified VDisk is mirrored.

-fmtdisk
(Optional) Specifies that the VDisk be formatted before use. This parameter formats the new extents that have been added to the VDisk as a result of the **expandvdisksize** command. The **expandvdisksize** command completes asynchronously if you use this parameter.

-unit b | kb | mb | gb | tb | pb
(Optional) Specifies the *disk_size* unit for the **-size** or **-rsize** parameter. The default value is megabytes (MB).

vdisk_name | vdisk_id
(Required) Specifies the virtual disk to modify, either by ID or by name.

Description

The **expandvdisksize** command can be used to expand the physical capacity that is allocated to a particular VDisk by the specified amount. The command can also be used to expand the virtual capacity of a space-efficient VDisk without altering the physical capacity that is assigned to the VDisk. To change the capacity of a non-space-efficient VDisk, or the virtual capacity of a space-efficient VDisk, use the **-size** parameter. To change the real capacity of a space-efficient VDisk, use the **-rsize** parameter.

The default capacity units are MB.

When a VDisk is expanded, the virtualization policy can change. Its mode becomes striped even if it was previously sequential. See the **svctask mkvdisk** command for details of the virtualization policies.

To run the **expandvdisksize** command on a mirrored VDisk, all copies of the VDisk must be synchronized. The command formats all copies of a mirrored VDisk automatically.

An invocation example

To increase the capacity of VDisk1 by 2048 bytes by using extents from two MDisks and to format the new part of the VDisk, enter:

```
svctask expandvdisksize -size 2048 -unit b -mdisk
mdisk0:mdisk1 -fmtdisk vdisk1
```

The resulting output

```
No feedback
```

An invocation example

To increase the capacity of VDisk1 by 100 MB using extents from two MDisks, and to format the new part of the VDisk, enter:

```
svctask expandvdisksize -size 100 -unit mb -mdisk mdisk0:mdisk1 -fmtdisk vdisk1
```

The resulting output

```
No feedback
```

An invocation example

To increase the real capacity of space-efficient VDisk **vdisk2** by 100 MB without changing the virtual capacity, and to spread the extents across all MDisks in the MDisk group, enter:

```
svctask expandvdisksize -rsize 100 -unit mb vdisk2
```

The resulting output

```
No feedback
```

An invocation example

To increase the real capacity of space-efficient VDisk copy id 1 of mirrored VDisk **vdisk3** by 100 MB, enter:

```
svctask expandvdisksize -rsize 100 -unit mb -copy 1 vdisk3
```

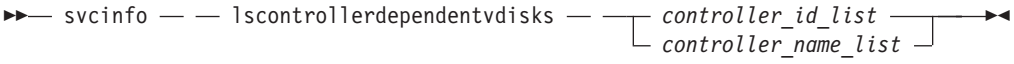
The resulting output

```
No feedback
```

lscontrollerdependentvdisks

The **lscontrollerdependentvdisks** command lists the VDisks that are dependent on the specified controller.

Syntax



Parameters

controller_id_list | *controller_name_list*
Specifies one or more controller IDs, controller names, or both. Separate multiple controllers using the colon character (**:**).

Description

The **lscontrollerdependentvdisks** command lists the VDisks that are dependent on the status of the specified controllers. If a controller goes offline, the dependent VDisks also go offline. Before taking a controller offline for maintenance, you can use the command to ensure that you do not lose access to any VDisks.

If you have multiple controllers configured as a single subsystem, you must specify all of the controllers in the subsystem, using a single command invocation.

The **lscontrollerdependentvdisks** command also checks for quorum disks on the specified controller list. If any quorum disks are on the specified controller list, the

command returns an error. All quorum disks must be moved before performing any maintenance. After moving quorum disks, reissue the command to list the dependent VDisks.

Note: The command lists the VDisks that are dependent on the controllers at the time the command is run; subsequent changes to your system require rerunning the command.

An invocation example

```
svcinfo lscontrollerdependentvdisks controller0
```

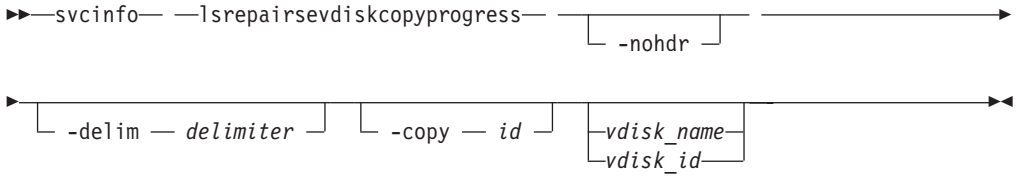
The resulting output

```
vdisk_id vdisk_name
0 vdisk0
1 vdisk1
2 vdisk2
```

lsrepairsevdiskcopyprogress

The **lsrepairsevdiskcopyprogress** command lists the repair progress for space-efficient VDisk copies.

Syntax



Parameters

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-copy *id*
(Optional) Lists the repair progress for the specified copy.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

vdisk_name | *vdisk_id*
(Optional) Specifies the virtual disk name or ID to list repair progress for. You

must specify this parameter last on the command line. If you do not enter this parameter, the command lists progress for all space-efficient copies in the cluster.

Description

The **lsrepairsevdiskcopyprogress** command lists the repair progress for space-efficient copies of the specified VDisk. If you do not specify a VDisk, the command lists the repair progress for all space-efficient copies in the cluster.

Note: Only run this command after running the **svctask repairsevdiskcopy** command, which you must only run as required by the Directed Maintenance Procedures or by IBM support.

An invocation example

```
svcinfo lsrepairsevdiskcopyprogress –delim :
```

The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:repairing:50:070301120000
0:vdisk0:1:repairing:51:070301120000
1:vdisk1:0:repairing:32:070301153500
```

An invocation example

```
svcinfo lsrepairsevdiskcopyprogress –delim : vdisk0
```

The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:repairing:50:070301120000
0:vdisk0:1:repairing:51:070301120000
```

An invocation example

```
svcinfo lsrepairsevdiskcopyprogress –delim : -copy 1 vdisk0
```

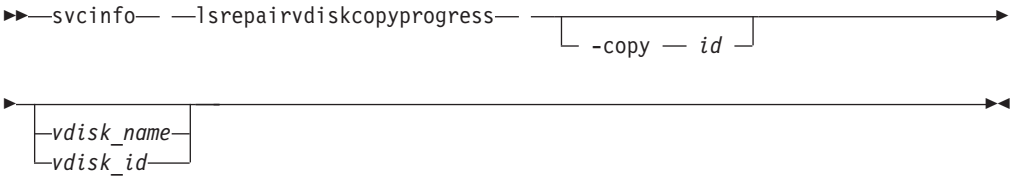
The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:1:repairing:51:070301120000
```

lsrepairvdiskcopyprogress

The **lsrepairvdiskcopyprogress** command displays the progress of mirrored VDisk repairs and validations.

Syntax



Parameters

-copy *id*
(Optional) Specifies the VDisk copy ID to list repair progress for. If you do not specify this parameter, progress is displayed for all copies.

vdisk_name | *vdisk_id*
(Optional) Specifies the virtual disk name or ID to list repair progress for. You must specify this parameter last on the command line.

Description

The **lsrepairvdiskcopyprogress** command displays the progress of repairs and validations being made to mirrored VDisks. Use this command to track progress after running the **svctask repairvdiskcopy** command. You can specify a VDisk copy using the **-copy** *id* parameter. To display the VDisks that have two or more copies with an active task, specify the command with no parameters; it is not possible to have only one VDisk copy with an active task.

The command displays progress for the following types of VDisk copies:

- All VDisk copies display the same task; validate, medium or resync, depending on the specified parameter.
- All VDisk copies display the same percentage and estimated completion time.
- If specified, non-mirrored VDisks are displayed as a single copy with a blank task; they are not displayed in the full concise view.
- Once a task completes, the task is blank for all copies.
- If the task is blank, the percentage and the completion time are also blank.

An invocation example

```
svcinfo lsrepairvdiskcopyprogress –delim :
```

The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:medium:50:070301120000
0:vdisk0:1:medium:50:070301120000
```

An invocation example

```
svcinfo lsrepairvdiskcopyprogress –delim : vdisk0
```

The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:medium:50:070301120000
0:vdisk0:1:medium:50:070301120000
```

An invocation example

```
svcinfo lsvdiskcopyrepairprogress –delim : -copy 0 vdisk0
```

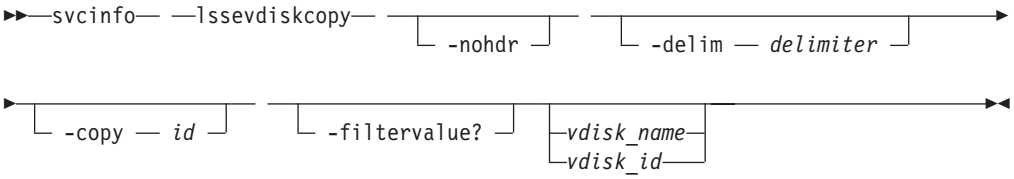
The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:medium:50:070301120000
```

lssevdiskcopy

The **lssevdiskcopy** command lists the space-efficient copies of the specified VDisks.

Syntax



Parameters

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-copy *id*
(Optional) Specifies the VDisk copy to list space-efficient copies for. You must specify a *vdisk_name* | *vdisk_id* value with this parameter.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-filtervalue?
(Optional) Displays a list of valid filter attributes. The following filters for the **svcinfo lssevdiskcopy** command are valid:

- mdisk_grp_id
- mdisk_grp_name
- overallocation
- autoexpand
- grainsize

vdisk_name | *vdisk_id*
(Optional) Specifies the virtual disk name or ID to list space-efficient copies for. You must specify this parameter last on the command line. If you do not enter this parameter, the command lists all space-efficient VDisk copies in the cluster.

Description

The **lssevdiskcopy** command lists all space-efficient copies of the specified VDisk. If you do not specify a VDisk, the command lists all space-efficient copies in the cluster.

The command provides a concise view of the space-efficient properties of the selected VDisk copies. Run the **svcinfo lsvdiskcopy** command to see a concise view of the properties that are common to space-efficient and non-space-efficient

VDisk copies. See the description of **svcinfo lsvdisk** command for a description of the fields that is shown in the view.

An invocation example

```
svcinfo lssevdiskcopy –delim :
```

The resulting output

```
vdisk_id:vdisk_name:copy_id:mdisk_grp_id:mdisk_grp_name:capacity:used_capacity:real
_capacity:
free_capacity:overallocation:autoexpand:warning:grainsize
0:vv1:0:0:ppp:16.0GB:5.0MB:4.0GB:15.99GB:400:off:20:32
1:se1:0:0:ppp:16.0GB:1.0GB:4.0GB:15.00GB:400:off:20:32
1:se1:1:0:ppp:16.0GB:2.0GB:8.0GB:14.00GB:200:off:45:256
```

An invocation example

```
svcinfo lssevdiskcopy –delim : se1
```

The resulting output

```
vdisk_id:vdisk_name:copy_id:mdisk_grp_id:mdisk_grp_name:capacity:used_capacity:real
_capacity:
free_capacity:overallocation:autoexpand:warning:grainsize
1:se1:0:0:ppp:16.0GB:1.0GB:4.0GB:15.00GB:400:off:20:32
1:se1:1:0:ppp:16.0GB:2.0GB:8.0GB:14.00GB:200:off:45:256
```

An invocation example

```
svcinfo lssevdiskcopy -delim : -copy 0 0
```

The resulting output

```
id:0
name:vdisk0
capacity:128.0MB
copy_id:0
status:online
sync:yes
primary:yes
mdisk_grp_id:0
mdisk_grp_name:mdiskgrp0
type:striped
mdisk_id:
mdisk_name:
fast_write_state:empty
used_capacity:0.41MB
real_capacity:128.00MB
free_capacity:127.59MB
overallocation:100
autoexpand:off
warning:79
grainsize:32
```

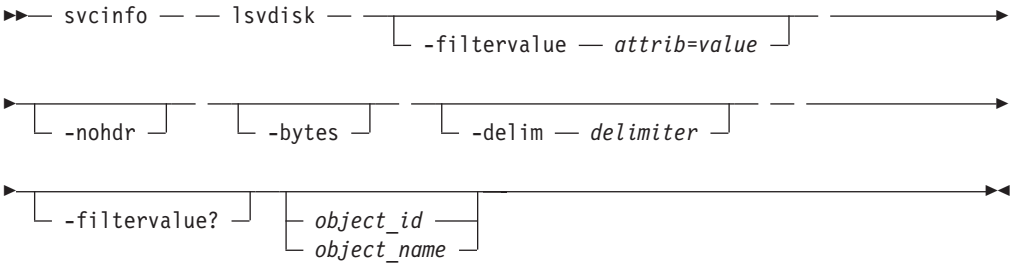
lsvdisk

The **lsvdisk** command displays a concise list or a detailed view of VDisks that are recognized by the cluster.

The list report style can be used to obtain two different styles of report.

- A list containing concise information about all the virtual disks that are recognized by the cluster. (Each entry in the list corresponds to a single virtual disk.)
- The detailed information about a single virtual disk.

Syntax



Parameters

-filtervalue *attrib=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are displayed. If a capacity is specified, the units must also be included.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-bytes
(Optional) Displays all capacities as bytes.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | object_name
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id | object_name* parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

-filtervalue?
(Optional) Displays a list of valid filter attributes. The following filters for the **svcinfo lsvdisk** command are valid:

- vdisk_name
- vdisk_id
- vdisk_UID
- fc_map_count
- copy_count
- IO_group_id

- IO_group_name
- status
- mdisk_grp_name
- mdisk_grp_id
- capacity
- type
- FC_id
- FC_name
- RC_id
- RC_name
- name
- id

Note: It is not possible to filter the **svcinfo lsvdisk** command with mdisk_grp_name=many to identify mirrored VDisks. Instead, filter on copy_count=2.

Description

This command displays a concise list or a detailed view of attributes for all VDisks and VDisk copies in the cluster.

The VDisk is offline and unavailable if one of the following take place:

- If both nodes in the I/O group are missing
- None of the nodes in the I/O group that are present can access the VDisk.
- All synchronized copies for this VDisk are in MDisk groups that are offline.

If you have a degraded VDisk and all of the associated nodes and MDisks are online, call the IBM Support Center for assistance. A VDisk is reported as degraded if any of the following occurs:

- One of the nodes in the I/O group is missing.
- One of the nodes in the I/O group cannot access all the MDisks in the MDisk group that the VDisk spans. In this case MDisks are shown as degraded and the DMPs for MDisks should be followed to resolve the problem.
- The fast write cache pins data for one or more VDisks in the I/O group and is unable to perform a failback until the situation is resolved. An error log indicating that the cache has pinned data is displayed. Follow the directed maintenance procedures for this error log to resolve the problem. The most common causes of pinned data are the following:
  - One or more VDisks in an I/O group is offline due to an asymmetric failure and has pinned data in the cache. Asymmetric failures can occur because of SAN fabric faults or misconfiguration, back-end controller faults or misconfiguration or because repeated errors has lead to the cluster excluding access to a MDisk through one or more nodes.
  - One or more VDisks in an I/O group is offline due to a problem with a FlashCopy mapping.

The command returns values for the following VDisk attributes:

IO_groups_id/name
Specifies the I/O Group that the VDisk belongs to.

status The value can be **online**, **offline** or **degraded**.

mdisk_grp_id/name
Specifies the name and ID of the MDisk group that the VDisk belongs to. If the VDisk has more than one copy, these fields display **many**.

type Specifies the virtualization type of the VDisk. The value can be **striped**, **sequential**, **image** or **many**. The value **many** indicates that the VDisk has more than one copy, which can have different virtualization types.

capacity
Specifies the total capacity of the VDisk.

formatted
Indicates whether the VDisk was formatted when it was created. The value can be **Yes** or **No**.

mdisk_id/name
Specifies the MDisk that is used for sequential and image mode VDisks. If the VDisk has more than one copy, these fields display **many**.

FC_id/name
Specifies the name and ID of the FlashCopy mapping that the VDisk belongs to. The value **many** indicates that the VDisk belongs to more than one FlashCopy mapping.

RC_id/name
Specifies the name and ID of the Global Mirror or Metro Mirror relationship that the VDisk belongs to.

vdisk_UID
Specifies the UID of the VDisk.

throttling
Specifies the throttle rate of the VDisk.

preferred_node_id
Specifies the ID of the preferred node for the VDisk.

fast_write_state
Specifies the cache state for the VDisk. The value can be **empty**, **not_empty**, **corrupt**, or **repairing**. A cache state of **corrupt** indicates that the VDisk requires recovery by using one of the **recovervdisk** commands. A cache state of **repairing** indicates that repairs initiated by a **recovervdisk** command are in progress.

cache Specifies the cache mode of the VDisk. The value can be **readwrite** or **none**.

udid Specifies the unit number for the VDisk. Only OpenVMS hosts require a unit number.

fc_map_count
Specifies the number of FlashCopy mappings that the VDisk belongs to.

sync_rate
Specifies the rate for synchronization for mirrored copies.

The command returns values for the following VDisk copy attributes:

copy_id
Specifies a system-assigned identifier for the VDisk copy. The value can be **0** or **1**.

status The value can be **online** or **offline**. A copy is offline if all nodes cannot access the MDisk group that contains the copy.

sync Indicates whether the VDisk copy is synchronized.

primary
Indicates whether the VDisk copy is the primary copy. A VDisk has exactly one primary copy. The value can be **Yes** or **No**.

mdiskgrp_id/name
Specifies the name and ID of the MDisk group that the VDisk copy belongs to.

type Specifies the virtualization type of the VDisk. The value can be **striped**, **sequential** or **image**.

mdisk_id/name
Specifies the MDisk that is used for sequential and image mode VDisks.

fast_write_state
Specifies the cache state of the VDisk copy. The value can be **empty**, **not_empty**, **corrupt**, or **repairing**. The value is always empty for non-space-efficient copies. A cache state of **corrupt** indicates that the VDisk is space-efficient and requires repair that is initiated by a **recovervdisk** command or the **repairsevdiskcopy** command.

used_capacity
Specifies the portion of **real_capacity** that is being used to store data. For non-space-efficient copies, this value is the same as the VDisk capacity. If the VDisk copy is space-efficient, the value increases from zero to the **real_capacity** value as more of the VDisk is written to.

real_capacity
Specifies the amount of physical storage that is allocated from an MDisk group to this VDisk copy. If the VDisk copy is not space-efficient, the value is the same as the VDisk capacity. If the VDisk copy is space-efficient, the value can be different.

free_capacity
Specifies the difference between the **real_capacity** and **used_capacity** values.

overallocation
Expressed as a percentage, specifies the ratio of VDisk capacity to **real_capacity** values. This value is always **100** for non-space-efficient VDisks.

autoexpand
Specifies whether **autoexpand** is enabled on a space-efficient VDisk. The value can be **on** or **off**.

warning
Expressed as a percentage, for space-efficient VDisk copies only. A warning is generated when the ratio of **used_capacity** to VDisk capacity reaches the specified level.

grainsize
For space-efficient VDisk copies, specifies the grain size chosen for the VDisk copy when it was created.

A concise invocation example

```
svcinfo lsvdisk -delim :
```

The concise resulting output

```
| id:name:IO_group_id:IO_group_name:status:mdisk_grp_id:mdisk_grp_name:capacity:type:FC_id:
|  FC_name:RC_id:RC_name:vdisk_UID:fc_map_count:copy_count:fast_write_state
| 0:vdisk0:0:io_grp0:degraded:0:mdiskgrp0:16.0MB:striped:::::60050768017F06BF7800000000000000:1
| 1:vdisk1:0:io_grp0:degraded:0:mdiskgrp0:16.0MB:striped:::::60050768017F06BF7800000000000001:1
| 2:vdisk2:0:io_grp0:degraded:0:mdiskgrp0:16.0MB:striped:::::60050768017F06BF7800000000000002:1
| 3:vdisk3:0:io_grp0:degraded:0:mdiskgrp0:16.0MB:striped:::::60050768017F06BF7800000000000003:1
```

A detailed invocation example

```
svcinfo lsvdisk -delim : 251
```

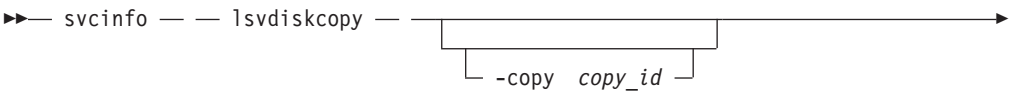
The detailed resulting output

```
id:251
name:i0vd163
IO_group_id:0
IO_group_name:io_grp0
status:online
mdisk_grp_id:3
mdisk_grp_name:vind1
capacity:16.0MB
type:striped
formatted:no
mdisk_id:
mdisk_name:
FC_id:
FC_name:
RC_id:
RC_name:
vdisk_UID:6005076801A0002C800000000000078B
throttling:0
preferred_node_id:1
fast_write_state:empty
cache:readwrite
udid:
fc_map_count:0
sync_rate:50
copy_count:1

copy_id:0
status:online
sync:yes
primary:yes
mdisk_grp_id:3
mdisk_grp_name:vind1
type:striped
mdisk_id:
mdisk_name:
fast_write_state:empty
used_capacity:16.00MB
real_capacity:16.00MB
free_capacity:0.00MB
overallocation:100
autoexpand:
warning:
grainsize:
```

lsvdiskcopy

The **lsvdiskcopy** command lists VDisk copy information.

Syntax



Parameters

-copy *copy_id*
(Optional) Specifies the VDisk copy to list information for. You must specify a *vdisk_name | vdisk_id* value with this parameter.

-filtervalue?
(Optional) Displays a list of valid filter attributes. The following filters for the **svcinfo lsvdiskcopy** command are valid:

- primary
- status
- sync
- mdisk_grp_id
- mdisk_grp_name
- type

vdisk_name | vdisk_id
(Optional) Specifies the VDisk to list copy information for. You must specify this parameter last on the command line. If you specify a *vdisk_name | vdisk_id* value only, all copies for the VDisk are listed.

Description

The **lsvdiskcopy** command lists information for VDisk copies. If you specify the command with no parameters, all VDisks and copies in the cluster are listed.

An invocation example

```
svcinfo lsvdiskcopy -delim :
```

The resulting output

```
vdisk_id:vdisk_name:copy_id:status:sync:primary:mdisk_grp_id:mdisk_grp_name:
capacity:type
0:vd1:0:online:yes:yes:1:mdisk_group_1:20GB:striped
0:vd1:1:offline:no:no:2:mdisk_group_2:20GB:striped
1:vd2:0:online:yes:yes:mdisk_group_2:100GB:image
```

An invocation example

```
svcinfo lsvdiskcopy -delim : vd1
```

The resulting output

```
vdisk_id:vdisk_name:copy_id:status:sync:primary:mdisk_grp_id:mdisk_grp_name:
capacity:type
0:vd1:0:online:yes:yes:1:mdisk_group_1:20GB:striped
0:vd1:1:offline:no:no:2:mdisk_group_2:20GB:striped
```

An invocation example

```
svcinfo lsvdiskcopy -copy 0 –delim : vv1
```

The resulting output

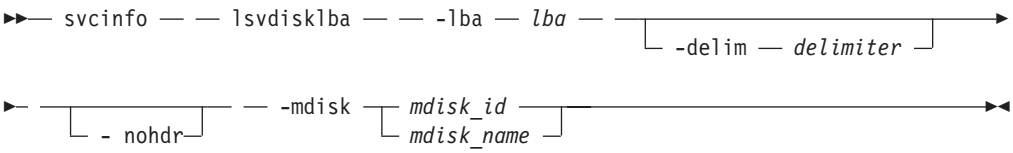
```
id:0
name:vdisk0
capacity:128.0MB
copy_id:0
```

```
status:online
sync:yes
primary:yes
mdisk_grp_id:0
mdisk_grp_name:mdiskgrp0
type:striped
mdisk_id:
mdisk_name:
fast_write_state:empty
used_capacity:0.41MB
real_capacity:128.00MB
free_capacity:127.59MB
overallocation:100
autoexpand:off
warning:79grainsize:32
```

lsvdisklba

The **lsvdisklba** command lists the VDisk and logical block address (LBA) for the specified MDisk LBA.

Syntax



Parameters

-lba *lba*
(Required) Specifies the 64-bit hexadecimal LBA on the MDisk.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-mdisk *mdisk_id* | *mdisk_name*
(Required) Specifies the MDisk name or ID.

Description

The **lsvdisklba** command returns one line of output that describes the LBA of the VDisk that is associated with the MDisk LBA.

If applicable, the command also lists the range of LBAs on both the VDisk and MDisk that are mapped in the same extent, or for space-efficient disks, in the same grain.

Table 5 provides command output that depends on several variables.

Table 5. lsvdisklba command output scenarios. Describes command output scenarios.

Field	Typical scenario	Quorum disk	Space-efficient metadata	Extent not allocated	Formatting extent	Extent allocated to space-efficient disk, LBA not used on space-efficient disk
copy_id	yes	no	yes	no	yes	yes
vdisk_id	yes	no	yes	no	yes	yes
vdisk_name	yes	no	yes	no	yes	yes
type	allocated	metadata	metadata	unallocated	formatting	unallocated
lba	yes	no	no	no	no	no
vdisk_start	yes	no	no	no	no	no
vdisk_end	yes	no	no	no	no	no
mdisk_start	yes	yes	yes	yes	yes	yes
mdisk_end	yes	yes	yes	yes	yes	yes

An invocation example

```
svcinfo lsvdisklba -mdisk 1 -lba 0x3480000 delim :
```

The resulting output

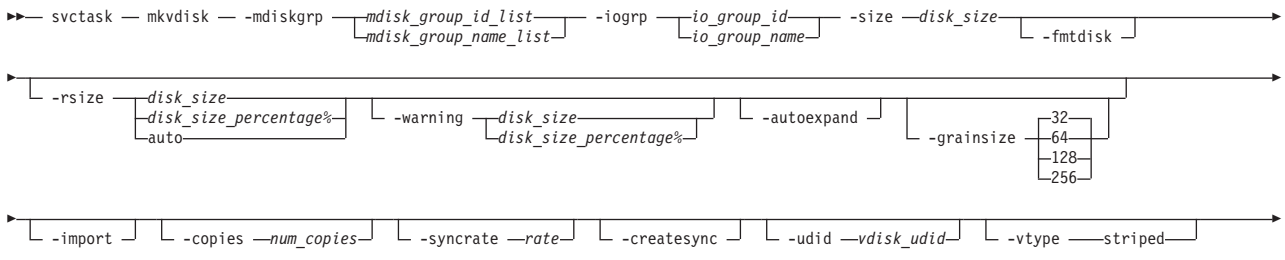
```
vdisk_id:vdisk_name:copy_id:type:lba:vdisk_start:vdisk_end:mdisk_start:mdisk_end
17:vdisk17:0:metadata:::0x3480000:0x34808ff
```

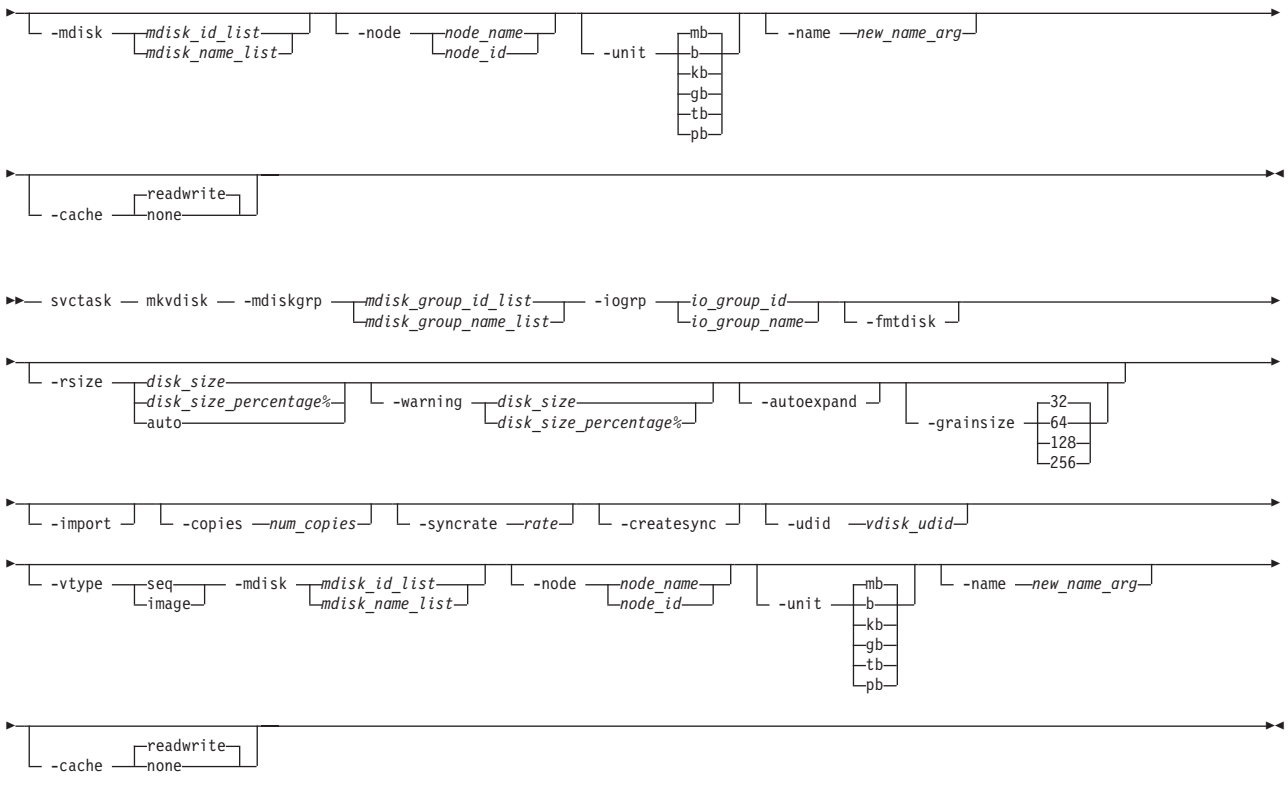
mkvdisk

The **mkvdisk** command creates sequential, striped, or image mode virtual disk objects. When they are mapped to a host object, these objects are seen as disk drives with which the host can perform I/O operations.

Note: The first syntax diagram depicts the creation of a **striped** mode virtual disk. The second syntax diagram depicts the creation of a **sequential** or **image** mode virtual disk.

Syntax





Parameters

-mdiskgrp *mdisk_group_id_list* | *mdisk_group_name_list*
(Required) Specifies one or more managed disk groups to use when you are creating this virtual disk. If you are creating multiple copies, you must specify one managed disk group per copy. The primary copy is allocated from the first managed disk group in the list.

-iogrp *io_group_id* | *io_group_name*
(Required) Specifies the I/O group (node pair) with which to associate this virtual disk.

-udid *vdisk_udid*
(Optional) Specifies the unit number (**udid**) for the disk. The **udid** is an identifier that is required to support OpenVMS hosts; no other systems use this parameter. Valid options are a decimal number 0 - 32 767, or a hexadecimal number 0 - 0x7FFF. A hexadecimal number must be preceded by **0x** (for example, **0x1234**).

-size *disk_size*
(Required for sequential [**seq**] or **striped** VDisk creation) (Optional for **image** VDisk creation) Specifies the capacity of the virtual disk, which is used with the value of the unit. All capacities, including changes, must be in multiples of 512 bytes. An error occurs if you specify a capacity that is not a multiple of 512, which can only happen when byte units (**-b**) are used. However, an entire extent is reserved even if it is only partially used. The default capacity is in MB. You can specify a capacity of 0. Specify the size in bytes in multiples of logical block address (LBA) sizes.

Note: If you do not specify the **-size** parameter when you create an image mode disk, the entire MDisk capacity is used.

-rsize *disk_size* | *disk_size_percentage***%** | **auto**
(Optional) Makes the VDisk space-efficient; otherwise, the VDisk is fully allocated. Specify the *disk_size* | *disk_size_percentage* value using an integer, or an integer immediately followed by the percent character (**%**). Specify the units for a *disk_size* integer using the **-unit** parameter; the default is MB. The **-rsize** value can be greater than, equal to, or less than the size of the VDisk. The **auto** option creates a VDisk copy that uses the entire size of the MDisk; if you specify the **-rsize auto** option, you must also specify the **-vtype image** option.

-warning *disk_size* | *disk_size_percentage***%**
(Optional) Requires that the **-rsize** parameter also be specified. Specifies a threshold at which a warning error log is generated for VDisk copies. A warning is generated when the used disk capacity on the space-efficient copy first exceeds the specified threshold. You can specify a *disk_size* integer, which defaults to MBs unless the **-unit** parameter is specified; or you can specify a *disk_size***%**, which is a percentage of the virtual disk size. If **-autoexpand** is enabled, the default value for **-warning** is 80% of the virtual disk capacity. If **-autoexpand** is not enabled, the default value for warning is 80% of the real capacity. To disable warnings, specify **0**.

-autoexpand
(Optional) Specifies that space-efficient copies automatically expand their real capacities by allocating new extents from their managed disk group. Requires that the **-rsize** parameter also be specified. If the **-autoexpand** parameter is specified, the **-rsize** parameter specifies a capacity that is reserved by the copy. This protects the copy from going offline when its managed disk group runs out of space by having the managed disk group to consume this reserved space first.

The parameter has no immediate effect on **image** mode copies. However, if the image mode copy is subsequently migrated to managed mode, the copy is then automatically expanded.

-grainsize 32 | 64 | 128 | 256
(Optional) Sets the grain size (KB) for a space-efficient VDisk. This parameter also requires that the **-rsize** parameter be specified. The default is 32 KB. If you are using the space-efficient VDisk in a FlashCopy map, use the same grain size as the map grain size for best performance. If you are using the space-efficient VDisk directly with a host system, use a small grain size.

-import
(Optional) Imports a space-efficient VDisk from the Mdisk. This parameter also requires that the **-rsize** parameter be specified.

-copies *num_copies*
(Optional) Specifies the number of copies to create. The *num_copies* value can be 1 or 2. Setting the value to 2 creates a mirrored VDisk. The default value is 1.

-syncrate *rate*
(Optional) Specifies the copy synchronization rate. A value of zero (**0**) prevents synchronization. The default value is **50**. For the supported **-syncrate** values and their corresponding rates, see Table 6 on page 110.

-createsync
(Optional) Creates copies in sync. Use this parameter if you have already formatted the MDisks, or when read stability to unwritten areas of the VDisk is not required.

-fmtdisk
(Optional) Specifies that the virtual disk be formatted before it can be used. The **-fmtdisk** parameter formats (sets to all zeros) the extents that make up this VDisk after it is created. If this parameter is used, the command completes asynchronously; you can query the status using the **svcinfo lsvdiskprogress** command.

The **-fmtdisk** parameter is not required when creating space-efficient virtual disks. Space-efficient VDisks return zeros for extents that have not been written to.

The **-fmtdisk** parameter synchronizes mirrored copies by default.

Note: You cannot specify this parameter with the **-vtype image** parameter.

-vtype seq | striped | image
(Optional) Specifies the virtualization type. When creating sequential or image mode VDisks, you must also specify the **-mdisk** parameter. The default virtualization type is striped.

-node *node_id | node_name*
(Optional) Specifies the preferred node ID or the name for I/O operations to this virtual disk. You can use the **-node** parameter to specify the preferred access node.

Note: This parameter is required for the subsystem device driver (SDD). The cluster chooses a default if you do not supply this parameter.

-unit b | kb | mb | gb | tb | pb
(Optional) Specifies the data units to use in conjunction with the capacity that is specified by the **-size** and **-rsize** parameters.

-mdisk *mdisk_id_list | mdisk_name_list*
(Optional) Specifies one or more managed disks. For sequential and image mode VDisks, the number of MDisks must match the number of copies. For sequential mode VDisks, each MDisk must belong to the specified MDisk group. For striped VDisks, you cannot specify the **-mdisk** parameter if the **-copies** value is greater than 1. When creating a single copy striped VDisk, you can specify a list of MDisks to stripe across.

-name *new_name_arg*
(Optional) Specifies a name to assign to the new virtual disk.

-cache readwrite | none
(Optional) Specifies the caching options for the VDisk. Valid entries are **readwrite** or **none**. The default is **readwrite**. If you do not specify the **-cache** parameter, the default value (**readwrite**) is used.

Description

This command creates a new virtual disk object. You can use the command to create a variety of types of virtual disk objects, making it one of the most complex commands.

You must decide which managed disk group or groups provide the storage for the VDisk. Use the **svcinfo lsmdiskgrp** command to list the available managed disk groups and the amount of free storage in each group. If you are creating a VDisk with more than one copy, each MDisk group that you specify must have enough space for the size of the VDisk.

Choose an I/O group for the VDisk. This determines which nodes in the cluster process the I/O requests from the host systems. If you have more than one I/O group, ensure that you distribute the VDisks between the I/O groups so that the I/O workload is shared evenly between all nodes. Use the **svcinfo lsiogrp** command to show the I/O groups and the number of virtual disks that are assigned to each I/O group.

Note: It is normal for clusters with more than one I/O group to have MDisk groups that have VDisks in different I/O groups. FlashCopy processing can make copies of VDisks whether the source and target VDisks are in the same I/O group. If, however, you plan to use intracluster Metro or Global Mirror operations, ensure that both the master and auxiliary VDisk are in the same I/O group.

Specify the virtualization type using the **-vtype** parameter; the supported types are sequential (**seq**), **striped**, and **image**.

sequential (seq)
This virtualization type creates the virtual disk using sequential extents from the specified MDisk (or MDisks, if creating multiple copies). The command fails if there are not enough sequential extents on the specified MDisk.

striped
This is the default virtualization type. If the **-vtype** parameter is not specified, **striped** is the default; all managed disks in the managed disk group are used to create the virtual disk. The striping is at an extent level; one extent from each managed disk in the group is used. For example, a managed disk group with 10 managed disks uses one extent from each managed disk, then it uses the 11th extent from the first managed disk, and so on.

If the **-mdisk** parameter is also specified, you can supply a list of managed disks to use as the stripe set. This can be two or more managed disks from the same managed disk group. The same circular algorithm is used across the striped set. However, a single managed disk can be specified more than once in the list. For example, if you enter `-mdisk 0:1:2:1`, the extents are from the following managed disks: 0, 1, 2, 1, 0, 1, 2, and so forth. All MDisks that are specified in the **-mdisk** parameter must be in the managed mode.

A capacity of 0 is allowed.

image This virtualization type allows image mode virtual disks to be created when a managed disk already has data on it, perhaps from a previrtualized subsystem. When an image mode virtual disk is created, it directly corresponds to the (previously unmanaged) managed disk that it was created from. Therefore, with the exception of space-efficient image mode VDisks, virtual disk logical block address (LBA) *x* equals managed disk LBA *x*. You can use this command to bring a nonvirtualized disk under the control of the cluster. After it is under the control of the cluster, you can migrate the virtual disk from the single managed disk. When it is migrated, the virtual disk is no longer an image mode virtual disk.

You can add image mode VDisks to an already populated MDisk group with other types of VDisks, such as a striped or sequential.

Note: An image mode VDisk must be 512 bytes or greater. At least one extent is allocated to an image mode VDisk.

You must use the **-mdisk** parameter to specify an MDisk that has a mode of unmanaged. The **-fmtdisk** parameter cannot be used to create an image mode VDisk.

Note: If you create a mirrored VDisk from two image mode MDisks without specifying a **-capacity** value, the capacity of the resulting VDisk is the smaller of the two MDisks, and the remaining space on the larger MDisk is not accessible.

The command returns the IDs of the newly created VDisk.

Attention:

1. Do not create a VDisk in an offline I/O group. You must ensure that the I/O group is online before you create a VDisk to avoid any data loss. This applies in particular to recreating VDisks that are assigned the same object ID.
2. To create an image mode disk, you must already have a quorum disk in the cluster because an image mode disk cannot be used to hold quorum data. See "Creating a quorum disk" in the *IBM System Storage SAN Volume Controller Software Installation and Configuration Guide* for more details.
3. The command fails if either limit of 2048 VDisks per I/O Group or 8192 VDisk copies per cluster is reached.

Table 6 provides the relationship of the *rate* value to the data copied per second.

Table 6. Relationship between the rate value and the data copied per second

User-specified *rate* attribute value	**Data copied/sec**
1 - 10	128 KB
11 - 20	256 KB
21 - 30	512 KB
31 - 40	1 MB
41 - 50	2 MB
51 - 60	4 MB
61 - 70	8 MB
71 - 80	16 MB
81 - 90	32 MB
91 - 100	64 MB

An invocation example

```
svctask mkvdisk -mdiskgrp Group0 -size 0
-iogrp 0 -vtype striped -mdisk mdisk1 -node 1
```

The resulting output

```
Virtual Disk, id [1], successfully created
```

An invocation example for creating an image mode VDisk

```
svctask mkvdisk -mdiskgrp Group0
-iogrp 0 -vtype image -mdisk mdisk2 -node 1
```

The resulting output

```
Virtual Disk, id [2], successfully created
```

An invocation example for creating a new VDisk

```
svctask mkvdisk -mdiskgrp Group0 -size 0 -unit kb
-iogrp 0 -vtype striped -mdisk mdisk1 -node 1 -udid 1234
```

The resulting output

```
Virtual Disk id [2], successfully created
```

An invocation example for creating a space-efficient VDisk

```
svctask mkvdisk -mdiskgrp Group0 -iogrp 0 -vtype striped
-size 10 -unit gb -rsize 20% -autoexpand -grainsize 32
```

The resulting output

```
Virtual Disk id [1], successfully created
```

An invocation example for creating a mirrored image-mode VDisk

```
svctask mkvdisk -mdiskgrp Group0:Group0 -mdisk mdisk2:mdisk3
-iogrp 0 -vtype image -copies 2
```

The resulting output

```
Virtual Disk id [1], successfully created
```

An invocation example for creating a mirrored VDisk

```
svctask mkvdisk -iogrp 0 -mdiskgrp 0:1 -size 500 -copies 2
```

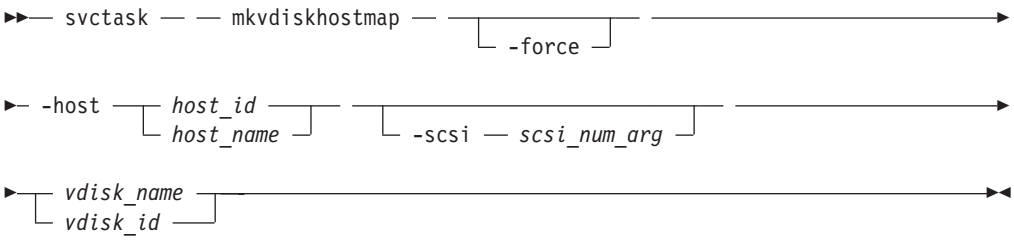
The resulting output

```
Virtual Disk id [5], successfully created
```

mkvdiskhostmap

The **mkvdiskhostmap** command creates a new mapping between a virtual disk and a host, which makes the virtual disk accessible for I/O operations to the specified host.

Syntax



Parameters

-force
(Optional) Allows multiple VDisk-to-host assignments, which are not normally allowed.

-host *host_id | host_name*
(Required) Specifies the host to map the virtual disk to, either by ID or by name.

-scsi *scsi_num_arg*
(Optional) Specifies the SCSI LUN ID to assign to this virtual disk on the given

host. The *scsi_num_arg* parameter contains the SCSI LUN ID that is assigned to the VDisk on the given host. You must check your host system for the next available SCSI LUN ID on the given HBA. If you do not specify the **-scsi** parameter, the next available SCSI LUN ID is provided to the host.

vdisk_name | vdisk_id
(Required) Specifies the name of the virtual disk that you want to map to the host, either by ID or by name.

Description

This command creates a new mapping between the virtual disk and the specified host. The virtual disk is presented to the host as if the disk is directly attached to the host. It is only after this command is processed, that the host can perform I/O transactions to the virtual disk.

Optionally, you can assign a SCSI LUN ID to the mapping. When the HBA in the host scans for devices that are attached to it, it discovers all virtual disks that are mapped to its fibre-channel ports. When the devices are found, each one is allocated an identifier (SCSI LUN ID). For example, the first disk found is usually SCSI LUN 1, and so on. You can control the order in which the HBA discovers virtual disks by assigning the SCSI LUN ID, as required. If you do not specify a SCSI LUN ID, the cluster automatically assigns the next available SCSI LUN ID, if any mappings already exist with that host.

Some HBA device drivers will stop when they find a gap in the SCSI LUN IDs. For example:

- Virtual Disk 1 is mapped to Host 1 with SCSI LUN ID 1
- Virtual Disk 2 is mapped to Host 1 with SCSI LUN ID 2
- Virtual Disk 3 is mapped to Host 1 with SCSI LUN ID 4

When the device driver scans the HBA, it must stop after identifying virtual disks 1 and 2, because no SCSI LUN is mapped with ID 3. For optimal performance, ensure that the SCSI LUN ID allocation is contiguous.

You can create multiple VDisk assignments. Normally, multiple VDisk-to-host assignments are not used because corruption is likely to occur if more than one host can access a disk. However, in certain multiple path environments, such as in the IBM SAN File System, a VDisk must be mapped to more than one host. To map to more than one host, you must use the **mkvdiskhostmap** command with the **-force** parameter. For example:

```
svctask mkvdiskhostmap -host host1 -force 4
svctask mkvdiskhostmap -host host2 -force 4
```

These commands create two host-to-VDisk mappings for VDisk 4 that map to host1 and host2. Omitting the **-force** parameter causes the mapping to fail if that VDisk is already mapped to a host.

The command also fails if the host object (to which this mapping is being made) is not associated with the I/O group containing the VDisk.

An invocation example

```
svctask mkvdiskhostmap -host host1 -scsi 1 5
```

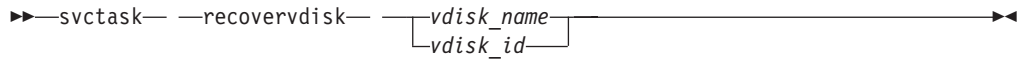
The resulting output

```
Virtual Disk to Host map, id [1], successfully created
```

recovervdisk

The **recovervdisk** command acknowledges VDisk data loss and brings the VDisk back online.

Syntax



Parameters

vdisk_name | *vdisk_id*
(Required) Specifies the virtual disk to recover.

Description

The specified VDisk, and all copies if mirrored, are recovered and brought back online. If the VDisk is space-efficient or has space-efficient copies, this command triggers the space-efficient repair process. If the VDisk is mirrored, the **recovervdisk** command triggers a resynchronization from a synchronized copy. The progress of the resynchronization can be monitored using the **svcinfo lsvdisksyncprogress** command. The VDisk remains online during the resynchronization process.

The **recovervdisk** command also starts the repair of any space-efficient copies that have a fast_write_state of corrupt. The progress of the repair process can be monitored using the **svcinfo lsrepairsevdiskcopyprogress** command.

A VDisk that is still offline because it is being repaired following the **recovervdisk** command has a fast_write_state of repairing. The VDisk is brought online when the repair process is complete.

An invocation example

```
svctask recovervdisk vdisk17
```

The resulting output

```
No feedback
```

recovervdiskbycluster

The **recovervdiskbycluster** command acknowledges data loss for all VDisks in the cluster with a fast_write_state of corrupt and brings the VDisks back online.

Syntax

Parameters

There are no parameters.

Description

All VDisks in the cluster that have a fast_write_state of corrupt; and all copies, if mirrored, are recovered and brought back online. If any of the VDisks are space-efficient or have space-efficient copies, the **recovervdiskbycluster** command triggers the space-efficient repair process. If VDisks are mirrored, the command triggers a resynchronization from a synchronized copy. The progress of the resynchronization can be monitored by using the **svctask lsvdisksyncprogress** command. VDisks remain online during the resynchronization process.

If none of the VDisks in the cluster have a fast_write_state of corrupt, the **recovervdiskbycluster** command still starts the repair process for any corrupt copies of mirrored VDisks. The progress of the repair process can be monitored using the **svcinfo lsrepairsevdiskcopyprogress** command. If there are no corrupt VDisks or no repairs to copies are required, no error is returned.

VDisks that are still offline because they are being repaired following the **recovervdiskbycluster** command have a fast_write_state of repairing. VDisks are brought online when the repair process is complete.

An invocation example

```
svctask recovervdiskbycluster
```

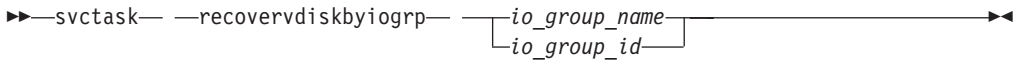
The resulting output

```
No feedback
```

recovervdiskbyiogrp

The **recovervdiskbyiogrp** command acknowledges data loss for all VDisks in the specified I/O group with a fast_write_state of corrupt and brings the VDisks back online.

Syntax



Parameters

io_group_name | *io_group_id*
(Required) Specifies the I/O group for virtual disk recovery.

Description

All VDisks in the specified I/O group that have a fast_write_state of corrupt; and all copies, if mirrored, are recovered and brought back online. If any of the VDisks are space_efficient or have space_efficient copies, the **recovervdiskbyiogrp** command triggers the space-efficient repair process. If VDisks are mirrored, the command triggers a resynchronization from a synchronized copy. The progress of the resynchronization can be monitored by using the **svctask lsvdisksyncprogress** command. VDisks remain online during the resynchronization process.

If none of the VDisks in the specified I/O group have a fast_write_state of corrupt, the **recovervdiskbyiogrp** command still starts the repair process for any corrupt

copies of mirrored VDisks. The progress of the repair process can be monitored using the **svcinfo lsrepairsevdiskcopyprogress** command. If there are no corrupt VDisks or no repairs to copies are required, no error is returned.

VDisks that are still offline because they are being repaired following the **recovervdiskbyiogrp** command have a fast_write_state of repairing. VDisks are brought online when the repair process is complete.

An invocation example

```
svctask recovervdiskbyiogrp iogrp2
```

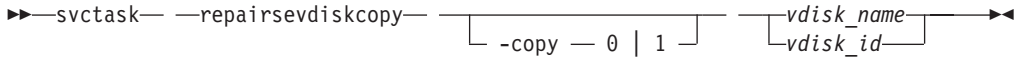
The resulting output

```
No feedback
```

repairsevdiskcopy

The **repairsevdiskcopy** command repairs the metadata on a space-efficient virtual disk.

Syntax



Parameters

-copy 0 | 1
(Optional) Specifies the VDisk copy to repair.

vdisk_name | vdisk_id
(Required) Specifies the virtual disk to repair.

Description

The **repairvsediskcopy** command repairs the metadata on a space-efficient VDisk. Run this command only when you are directed by the Directed Maintenance Procedures or by IBM support.

Running the command automatically detects corrupted metadata. The command holds the VDisk offline during the repair, but does not prevent the disk from being moved between I/O groups.

If a repair operation completes successfully and the volume was previously offline because of corrupted metadata, the command brings the volume back online. The only limit on the number of concurrent repair operations is the number of virtual disk copies in the configuration. Once started, a repair operation cannot be paused or canceled; the repair can only be ended by deleting the copy.

An invocation example

```
svctask repairsevdiskcopy vdisk8
```

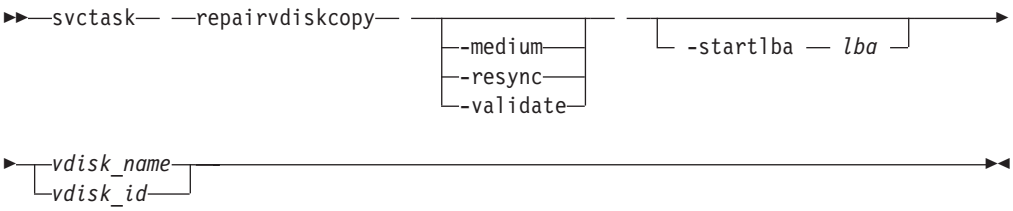
The resulting output

```
No feedback
```

repairvdiskcopy

The **repairvdiskcopy** command detects and optionally, corrects any VDisk copies that are not identical.

Syntax



Parameters

-medium
(Optional) Converts sectors that contain different contents into virtual medium errors on the specified VDisk. This parameter cannot be used with the **-validate** and **-resync** parameters; you must enter one of the three parameters.

-resync
(Optional) Corrects sectors that contain different contents by copying contents from the primary VDisk copy to other copies on the specified VDisk. This parameter cannot be used with the **-medium** and **-resync** parameters; you must enter one of the three parameters.

-validate
(Optional) Reports the first difference found on synchronized online copies of the specified VDisk, on or after the specified **-startlba** value. This parameter cannot be used with the **-medium** and **-resync** parameters; you must enter one of the three parameters.

-startlba *lba*
(Optional) Specifies a starting Logical Block Address (LBA) on which to begin the command.

vdisk_name | *vdisk_id*
(Required) Specifies the virtual disk to repair. You must specify this parameter last on the command line.

Description

The **repairvdiskcopy** command detects and optionally, corrects any VDisk copies that are not identical. The results are logged to the SAN Volume Controller error log. The **-validate** parameter compares synchronized online copies of the specified VDisk. The **-medium** parameter changes any sectors that are not identical into virtual medium errors. The **-resync** parameter copies any sectors that are not identical to the other VDisk copies. You must specify only one of the three parameters.

Attention:

1. Before you run the **repairvdiskcopy** command, ensure that all VDisk copies are synchronized.
2. Only one **repairvdiskcopy** command can run on a VDisk at a time. You must wait for the **repairvdiskcopy** command to complete processing before running the command again.
3. Once you start the **repairvdiskcopy** command, you cannot use the command to stop processing.
4. The primary copy of a mirrored VDisk cannot be changed while the **repairvdiskcopy -resync** command is running.

Use the **-startlba** parameter to specify a starting Logical Block Address (LBA). Enter an LBA value from 0 - full disk size minus one. The parameter logs the first error found and then stops the command. By repeating this parameter, you can collect all of the instances where the VDisk copies are not identical.

During **repairvdiskcopy** command operation, the VDisk remains online. The I/O and synchronization operations are allowed while the command is in progress.

The rate for the **repairvdiskcopy** command is controlled by the synchronization rate of the VDisk that is being repaired. To suspend the repair process, set the synchronization rate of the VDisk to **0** using the **chvdisk** command.

An invocation example

```
svctask repairvdiskcopy -resync vdisk8
```

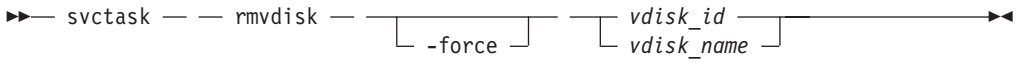
The resulting output

```
No feedback
```

rmvdisk

The **rmvdisk** command deletes a virtual disk (VDisk).

Syntax



Parameters

-force
(Optional) Deletes the specified VDisk, even if mappings still exist between this virtual disk and one or more hosts. This parameter deletes any host-to-VDisk mappings and any FlashCopy mappings that exist for this VDisk. This parameter also deletes any Metro Mirror or Global Mirror relationships that exist for the specified VDisk. Data that is on the virtual disk is lost. Before you issue this command, ensure that the virtual disk and any data that resides on it are no longer required.

vdisk_id | *vdisk_name*
Specifies the name of the virtual disk to delete, either by ID or by name.

Description

This command deletes an existing managed mode virtual disk or an existing image mode virtual disk. The extents that made up this virtual disk are returned to the pool of free extents that are available on the managed disk group, if the VDisk is in managed mode.

Attention: Any data that was on the virtual disk is lost. Before you issue this command, ensure that the virtual disk (and any data that resides on it) is no longer required.

Deleting a managed mode virtual disk

When you use this command to delete a managed mode virtual disk, all the data on the virtual disk is deleted. The extents that make up the virtual disk are returned to the pool of free extents that are available in the managed disk group.

If host mappings exist for the virtual disk, or if any FlashCopy mappings would be affected, the deletion fails. You can use the **-force** parameter to force the deletion. If you use the **-force** parameter, mappings that have the virtual disk as source or target are deleted, other mappings in a cascade might be stopped, and then the virtual disk is deleted. The **-force** parameter also deletes any Metro Mirror or Global Mirror relationships that exist for the specified VDisk.

If the virtual disk is in the process of migrating to an image mode virtual disk (using the **svctask migratetoimage** command), the deletion fails unless you use the **-force** parameter. If you use the **-force** parameter, the migration is halted and then the virtual disk is deleted. Before you issue this command, ensure that the virtual disk (and any data that resides on it) is no longer required.

Deleting an image mode virtual disk

If the VDisk is mirrored and one or both copies is in image mode, you must first wait for all fast-write data to be moved to the controller logical unit. This ensures that the data on the controller is consistent with the data on the image mode virtual disk before the VDisk is deleted. This process can take several minutes to complete, and is indicated by the *fast_write_state* state of the virtual disk being **empty**. If the **-force** parameter is specified, the fast-write data is discarded and the virtual disk is deleted immediately; the data on the controller logical unit is left inconsistent and unusable. If the copies are not synchronized, you must use the **-force** parameter.

If you run the command while data is in the cache, SVC attempts to move the data out of the cache; this process can time out, however.

If there are any virtual medium errors on the virtual disk, the command fails. You can force the deletion by using the **-force** parameter; however, this can cause data integrity problems.

Note: A virtual medium error occurs when you copy data from one disk (the source) to another (the target). Reading the source indicates that there is a medium error. At that moment, you must have two identical copies of data and you must then simulate a medium error on the target disk. You can simulate a medium error on the target disk by creating a virtual medium error on the target disk.

If FlashCopy mappings or host mappings exist for the virtual disk, the deletion fails unless you use the **-force** parameter. If you use the **-force** parameter, mappings are deleted and the virtual disk is deleted. If there is any data that is not staged in the fast write cache for this virtual disk, the deletion of the virtual disk fails. When the **-force** parameter is specified, any data that is not staged in the fast write cache is deleted. Deleting an image mode virtual disk causes the managed disk that is associated with the virtual disk to be removed from the managed disk group. The mode of the managed disk is returned to "unmanaged."

An invocation example

```
svctask rmvdisk -force vdisk5
```

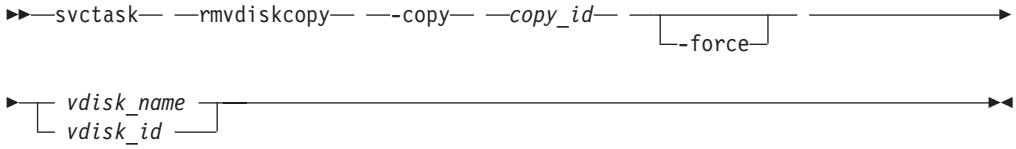
The resulting output

```
No feedback
```

rmvdiskcopy

The **rmvdiskcopy** command removes a VDisk copy from a VDisk.

Syntax



Parameters

-copy *copy_id*
(Required) Specifies the ID of the copy to delete.

-force
(Optional) Forces the deletion of the last synchronized copy of a VDisk, which deletes the entire VDisk. The parameter also forces the deletion of a nonmirrored VDisk, a copy that is migrating to image mode, or an image-mode copy that has virtual medium errors.

vdisk_name | *vdisk_id*
(Required) Specifies the virtual disk to delete the copy from. You must specify this parameter last on the command line.

Description

The **rmvdiskcopy** command deletes the specified copy from the specified VDisk. The command fails if all other copies of the VDisk are not synchronized; in this case, you must specify the **-force** parameter, delete the VDisk, or wait until the copies are synchronized.

An invocation example

```
svctask rmvdiskcopy -copy 1 vdisk8
```

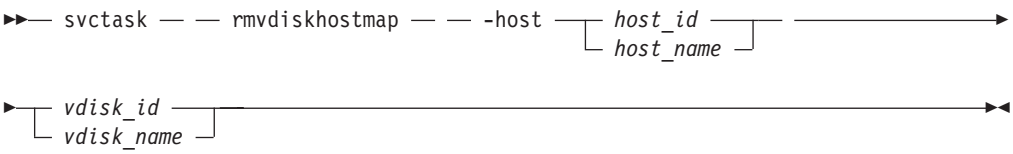
The resulting output

```
No feedback
```

rmvdiskhostmap

The **rmvdiskhostmap** command deletes an existing virtual disk-to-host mapping; the virtual disk is no longer accessible for I/O transactions on the given host.

Syntax



Parameters

-host *host_id* | *host_name*
(Required) Specifies the host that you want to remove from the map with the virtual disk, either by ID or by name.

vdisk_id | *vdisk_name*
(Required) Specifies the name of the virtual disk that you want to remove from the host mapping, either by ID or by name.

Description

This command deletes an existing mapping between the specified virtual disk and the host. This effectively stops the virtual disk from being available for I/O transactions on the given host.

This command also deletes a SCSI or persistent reservation that a host has on a VDisk. Once the reservation is removed, a new host is allowed to access the VDisk in the future because the original host no longer has access.

Use caution when you process this command because to the host, it seems as if the virtual disk has been deleted or is offline.

An invocation example

```
svctask rmvdiskhostmap -host host1 vdisk8
```

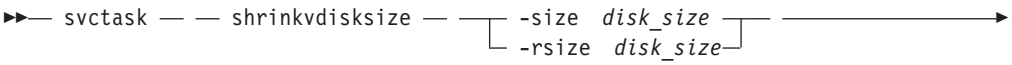
The resulting output

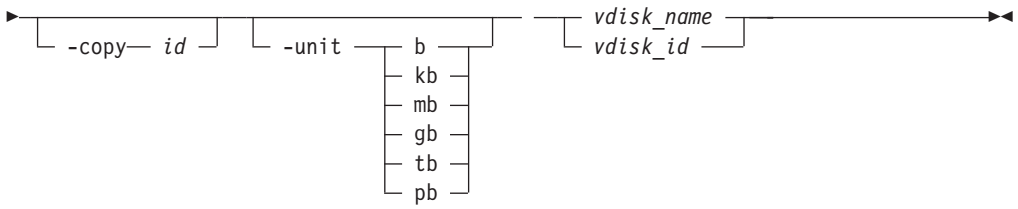
```
No feedback
```

shrinkvdisksize

The **shrinkvdisksize** command reduces the size of a VDisk by the specified capacity.

Syntax





Parameters

-size *disk_size*
(Required) Specifies the size reduction for the designated virtual disk. The **-size** parameter cannot be used with the **-rsize** parameter. You must specify either **-size** or **-rsize**.

-rsize *disk_size*
(Optional) Reduces the real size of a space-efficient VDisk by the specified amount. Specify the *disk_size* value using an integer. Specify the units for a *disk_size* integer using the **-unit** parameter; the default is MB. The **-rsize** value can be greater than, equal to, or less than the size of the VDisk. You must specify either the **-size** parameter or the **-rsize** parameter.

-copy *id*
(Optional) Specifies the copy to change the real capacity for. You must also specify the **-rsize** parameter. If the **-copy** parameter is not specified, all copies of the VDisk are reduced. This parameter is required if the VDisk is mirrored and only one copy is space-efficient.

-unit b | kb | mb | gb | tb | pb
(Optional) Specifies the data units to be used in conjunction with the value that is specified by the **-size** parameter.

vdisk_name | *vdisk_id*
(Required) Specifies the virtual disk that you want to modify, either by ID or by name.

Description

The **shrinkvdisksize** command reduces the capacity that is allocated to the particular virtual disk by the amount that you specify. You cannot shrink the real size of a space-efficient volume below its used size. All capacities, including changes, must be in multiples of 512 bytes. An entire extent is reserved even if it is only partially used. The default capacity units are MB.

The command can be used to shrink the physical capacity that is allocated to a particular VDisk by the specified amount. The command can also be used to shrink the virtual capacity of a space-efficient VDisk without altering the physical capacity assigned to the VDisk. To change the capacity of a non-space-efficient disk, use the **-size** parameter. To change the real capacity of a space-efficient disk, use the **-rsize** parameter. To change the virtual capacity of a space-efficient disk, use the **-size** parameter.

VDisks can be reduced in size, if required.

When the virtual size of a space-efficient VDisk is changed, the warning threshold is automatically scaled to match. The new threshold is stored as a percentage.

To run the **shrinkvdisksize** command on a mirrored VDisk, all copies of the VDisk must be synchronized.

Attention: If the VDisk contains data that is being used, do not shrink the VDisk without backing up the data first.

The cluster arbitrarily reduces the capacity of the VDisk by removing a partial, one or more extents from those allocated to the VDisk. You cannot control which extents are removed and so you cannot assume that it is unused space that is removed.

Attention:

1. If the virtual disk contains data, do not shrink the disk.
2. Some operating systems or file systems use what they consider to be the outer edge of the disk for performance reasons. This command can shrink FlashCopy target virtual disks to the same capacity as the source.
3. Before you shrink a VDisk, validate that the VDisk is not mapped to any host objects. If the VDisk is mapped, data is displayed. You can determine the exact capacity of the source or master VDisk by issuing the **svcinfo lsvdisk -bytes** *vdiskname* command. Shrink the VDisk by the required amount by issuing the **svctask shrinkvdisksize -size** *disk_size* **-unit b | kb | mb | gb | tb | pb** *vdisk_name | vdisk_id* command.

An invocation example

```
svctask shrinkvdisksize -size 2048 -unit b vdisk1
```

The resulting output

```
No feedback
```

An invocation example

To decrease the capacity of **vdisk1** by 100 MB, enter:

```
svctask shrinkvdisksize -rsize 1024 -unit b vdisk2
```

The resulting output

```
No feedback
```

An invocation example

To decrease the real capacity of space-efficient VDisk **vdisk2** by 100 MB without changing its virtual capacity, enter:

```
svctask shrinkvdisksize -rsize 100 -unit mb vdisk2
```

The resulting output

```
No feedback
```

An invocation example

To decrease the real capacity of space-efficient VDisk copy ID 1 of mirrored VDisk **vdisk3** by 100 MB, enter:

```
svctask shrinkvdisksize -rsize 100 -unit mb -copy 1 vdisk3
```

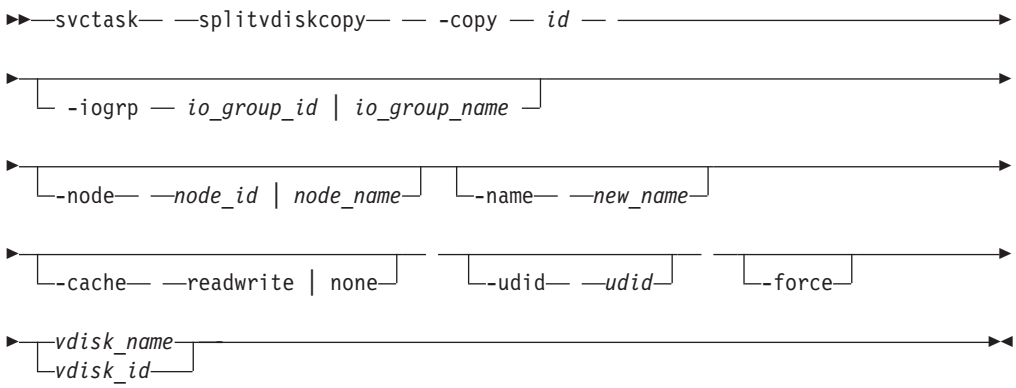
The resulting output

```
No feedback
```

splitvdiskcopy

The **splitvdiskcopy** command creates a separate VDisk from a synchronized copy of a mirrored VDisk.

Syntax



Parameters

-copy *id*
(Required) Specifies the ID of the copy to split.

-iogrp *io_group_id | io_group_name*
(Optional) Specifies the I/O group to add the new virtual disk to. The default is the I/O group of the specified VDisk.

-node *node_id | node_name*
(Optional) Specifies the preferred node ID or the name for I/O operations to this virtual disk. You can use the **-node** parameter to specify the preferred access node.

-name *new_name*
(Optional) Assigns a name to the new virtual disk.

-cache readwrite | none
(Optional) Specifies the caching options for the new virtual disk. Enter **readwrite** or **none**; the default is **readwrite**.

-udid *udid*
(Optional) Specifies the *udid* for the new VDisk. The *udid* is a required identifier for OpenVMS hosts; no other hosts use this parameter. Supported values are a decimal number 0 - 32 767, or a hexadecimal number 0 - 0x7FFF. A hexadecimal number must be preceded by **0x**; for example, **0x1234**. The default *udid* value is **0**.

-force
(Optional) Allows the split to proceed even when the specified copy is not synchronized, or even when the cache flush is likely to fail. The newly created VDisk might not be consistent.

Description

The **splitvdiskcopy** command creates a new VDisk in the specified I/O Group from a copy of the specified VDisk. If the copy that you are splitting is not synchronized, you must use the **-force** parameter. The command fails if you are

attempting to remove the only synchronized copy. To avoid this, wait for the copy to synchronize or split the unsynchronized copy from the VDisk by using the **-force** parameter. You can run the command when either VDisk copy is offline.

An invocation example

```
svctask splitvdiskcopy -copy 1 vdisk8
```

The resulting output

```
Virtual Disk, id [1], successfully created.
```

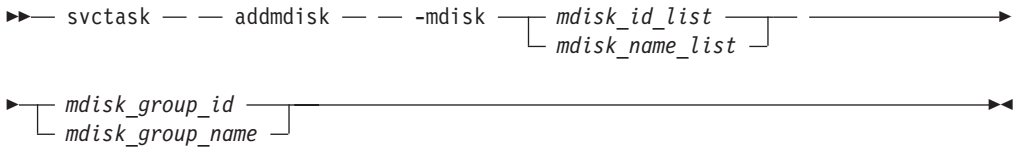
Chapter 12. Managed disk group commands

The following commands enable you to work with managed disk group options with the SAN Volume Controller.

addmdisk

The **addmdisk** command adds one or more managed disks to an existing managed disk group.

Syntax



Parameters

-mdisk *mdisk_id_list* | *mdisk_name_list*
(Required) Specifies one or more managed disk IDs or names to add to the group.

mdisk_group_id | *mdisk_group_name*
(Required) Specifies the ID or name of the managed disk group to add the disks to. When an MDisk is added, the warning threshold for the MDisk group is automatically scaled.

Description

This command adds the managed disks that you specify to the group. The disks can be specified in terms of the managed disk ID or the managed disk name.

The managed disks must be in unmanaged mode. Disks that already belong to a group cannot be added to another group until they have been deleted from their current group. You can delete a managed disk from a group under the following circumstances:

- If the managed disk does not contain any extents in use by a virtual disk
- If you can first migrate the extents in use onto other free extents within the group.

An invocation example

```
svctask addmdisk -mdisk mdisk13:mdisk14 Group0
```

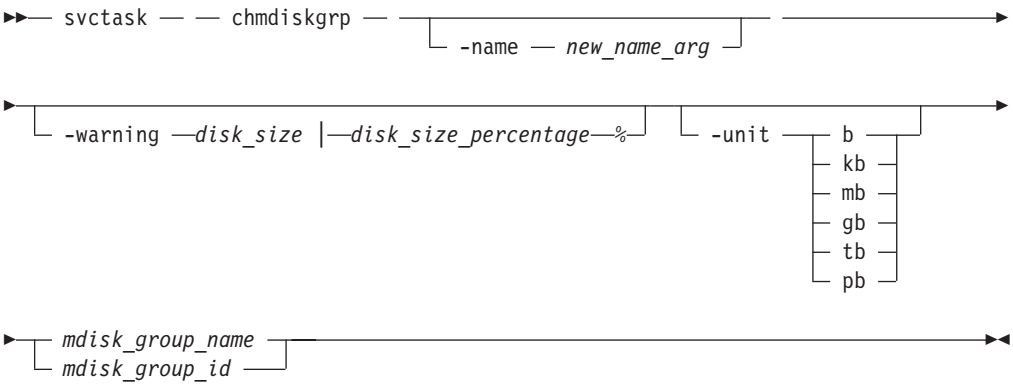
The resulting output

```
No feedback
```

chmdiskgrp

Use the **chmdiskgrp** command to modify the name that is assigned to a managed disk (MDisk) group or to set the warning threshold for the MDisk group.

Syntax



Parameters

-name *new_name_arg*
(Optional) Specifies the new name of the managed disk group.

-warning *disk_size* | *disk_size_percentage***%**
(Optional) Sets a threshold at which a warning is generated. The warning is generated the first time that the threshold is exceeded by the used-disk capacity in the MDisk group. You can specify a *disk_size* integer, which defaults to megabytes (MB) unless the **-unit** parameter is specified; or you can specify a *disk_size***%**, which is a percentage of the MDisk group size. To disable warnings, specify **0** or **0%**.

-unit b | kb | mb | gb | tb | pb
(Optional) Specifies the data units for the **-warning** parameter.

mdisk_group_id | *mdisk_group_name*
(Required) Specifies the ID or name of the managed disk group to modify.

Description

This command modifies the name, or label, assigned to a given managed disk group. Subsequently, you can use the new name to refer to the managed disk group.

The command can also be used to set the warning threshold for the managed disk group. The warning threshold is the threshold at which a warning is generated when it is exceeded by the used-disk capacity in the MDisk group.

An invocation example

```
svctask chmdiskgrp -name testmdiskgrp Group0
```

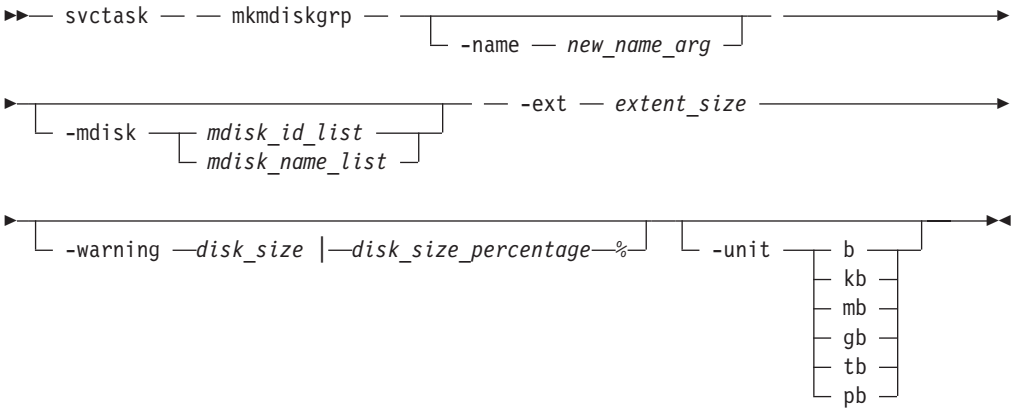
The resulting output

```
No feedback
```

mkmdiskgrp

The **mkmdiskgrp** command creates a new managed disk group.

Syntax



Parameters

-name *new_name_arg*
(Optional) Specifies a name to assign to the new group.

-mdisk *mdisk_id_list* | *mdisk_name_list*
(Optional) Specifies a colon-separated list of managed disk IDs or names to add to the group. You can create an empty MDisk group by not specifying the **-mdisk** parameter.

-ext *extent_size*
(Required) Specifies the size of the extents for this group in MB. The *extent_size* parameter must be one of the following values: **16**, **32**, **64**, **128**, **256**, **512**, **1024**, or **2048** (MB).

-warning *disk_size* | *disk_size_percentage***%**
(Optional) Generates a warning when the used disk capacity in the MDisk group first exceeds the specified threshold. You can specify a *disk_size* integer, which defaults to megabytes (MB) unless the **-unit** parameter is specified; or you can specify a *disk_size***%**, which is a percentage of the MDisk group size. To disable warnings, specify **0** or **0%**. The default value is **0**.

-unit b | kb | mb | gb | tb | pb
(Optional) Specifies the data units for the **-warning** parameter.

Description

The **mkmdiskgrp** command creates a new managed disk group and assigns the group name if specified. The ID of the new group is returned if the command is successful. Managed disk groups are collections of managed disks. Each group is divided into chunks, called extents, which are used to create virtual disks.

Optionally, you can specify a list of managed disks that will be added to this group. These managed disks cannot belong to another group, and they must have a mode of unmanaged. Use the **svcinfo lsmdiskcandidate** command to get a list of suitable candidates.

Each managed disk that is a member of this group is split into extents. The storage that is available on these disks is added to a pool of extents that is available in this

group. When a virtual disk is created from this group, free extents from the pool are used, in accordance with the policy used when the virtual disk was first created.

All managed disks subsequently added to this group are split into extents of the same size as the size that is assigned to the group.

When choosing an extent size, take into account the amount of storage you want to virtualize in this group. The system maintains a mapping of extents between virtual disks and managed disks. The cluster can only manage a finite number of extents (4 194 304). One cluster can virtualize the following number of extents:

- 64 TB – if all managed disk groups have extent sizes of 16 MB.
- 2 PB – if all managed disk groups have extent sizes of 512 MB.

Note: When an image mode VDisk is created, the MDisk group increases in capacity by the size of the image mode VDisk (not the MDisk capacity), because the image mode VDisk might be smaller than the MDisk itself. If an extent is migrated from the image mode VDisk or MDisk to elsewhere in the group, the VDisk becomes a striped VDisk (no longer image mode). At this point the available capacity might increase, because the extra capacity available on the MDisk (for example, the capacity that was not part of the image mode VDisk) becomes available.

An invocation example

```
svctask mkmdiskgrp -mdisk mdisk13 -ext 512
```

The resulting output

```
MDisk Group, id [1], successfully created
```

An invocation example

```
svctask mkmdiskgrp -mdisk mdisk0:mdisk1:mdisk2:mdisk3 -ext 32
```

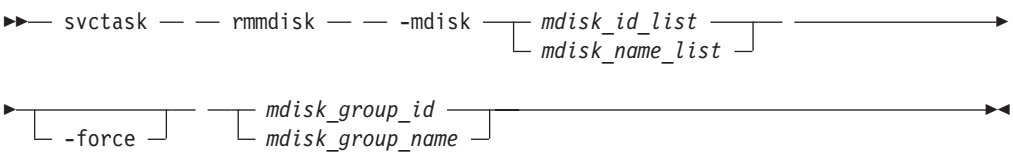
The resulting output

```
MDisk Group, id [0], successfully created
```

rmmdisk

The **rmmdisk** command deletes a managed disk (MDisk) from a managed disk group.

Syntax



Parameters

-mdisk *mdisk_id_list | mdisk_name_list*
(Required) Specifies one or more managed disk IDs or names to delete from the group.

-force
(Optional) Migrates data on the specified disks to other disks in the group. The command completes asynchronously if **-force** is specified.

mdisk_group_id | mdisk_group_name
(Required) Specifies the ID or name of the managed disk group to delete the disks from. The warning threshold for an MDisk group is automatically scaled when MDisks are deleted.

Description

This command attempts to remove the managed disk or disks from the group.

Deleting a managed disk from a group can only be done if the managed disk does not contain any extents in use by a virtual disk. If there are extents in use and you do not supply the force flag, the command fails.

Attention: If this disk being removed has already been powered down, removed, or is experiencing a power outage, the migration is pending and does not complete until the MDisk comes back online. The MDisk is not removed from the list of MDisks that are contained in the group.

If the disk has been deliberately removed, the only method of removing the MDisk is to remove the entire group itself.

Ensure that you do not destroy any controller LUNs until you have deleted them from the MDisk group that they belong to.

The **rmmdisk** command fails if there are insufficient free extents on other disks in the mdisk group for the duration of the command. To avoid this problem, do not issue new commands that use extents until **rmmdisk** processing is completed.

If you do specify the force flag, an attempt will be made to migrate the extents that are in use onto other free extents within the group. If there are not enough free extents in the group, the command will fail even if the force flag is specified.

To delete the disks from the group, you have the following options:

- You can delete the virtual disk that is using the extents specified on the managed disk.
- You can add more managed disks to the group, rerun the command and specify the **-force** parameter.

When data is being migrated from the managed disk, it might take some time for the command to complete. The command itself will return with a success code, notifying you that migration is in progress. An event is logged when the migration is complete and the disk is deleted from the group at this time. You can also check the progress of any active migrations by running the **svcinfo lsmigrate** command.

If the **-force** parameter is used, the **rmmdisk** command fails if the target or source VDisk is offline, or if there is insufficient quorum-disk space to store the metadata. Correct the offline or quorum disk condition and try reissuing the command.

An invocation example

```
svctask rmmdisk -mdisk mdisk12 -force Group3
```

The resulting output

rmmdiskgrp

The **rmmdiskgrp** command deletes a managed disk group so that there is no possibility to recover it.

Syntax

Parameters

-force
(Optional) Specifies that all virtual disks and virtual disk-to-host mappings be deleted.

Attention: Use this parameter with extreme caution. When you use this parameter, all managed disks in the group are removed and the group itself is deleted.

mdisk_group_id | *mdisk_group_name*
(Required) Specifies the ID or name of the managed disk group that is to be deleted.

Description

The **rmmdiskgrp** command deletes the specified managed disk group. The **-force** parameter is required if there are virtual disks that have been created from this group or if there are managed disks in the group. Otherwise, the command fails.

Deleting a managed disk group is essentially the same as deleting a cluster or part of a cluster, because the managed disk group is the central point of control of virtualization. Because virtual disks are created using available extents in the group, mapping between virtual disk extents and managed disk extents is controlled based on the group.

The command deletes all VDisk copies in the specified MDisk group. If the VDisk has no remaining synchronized copies in other MDisk groups, the VDisk is also deleted.

Attention:

1. This command partially completes asynchronously. All virtual disks, host mappings, and Copy Services relationships are deleted before the command completes. The deletion of the managed disk group then completes asynchronously.
2. Before you issue the command, ensure that you want to delete all mapping information; data that is contained on virtual disks cannot be recovered after the managed disk group has been deleted.

In detail, if you specify the **-force** parameter and the virtual disks are still using extents in this group, the following actions are initiated or occur:

- The mappings between that disk and any host objects and the associated Copy Services relationships are deleted.

- If the virtual disk is a part of a FlashCopy mapping, the mapping is deleted.

  Note: If the mapping is not in the **idle_or_copied** or **stopped** states, the mapping is force-stopped and then deleted. Force-stopping the mapping might cause other FlashCopy mappings in the cluster to also be stopped. See the description for the **-force** parameter in the **stopfcmap** command for additional information.
- Any virtual disk that is in the process of being migrated into or out of the managed disk group is deleted. This frees up any extents that the virtual disk was using in another managed disk group.
- Virtual disks are deleted without first flushing the cache. Therefore, the storage controller LUNs that underlie any image mode MDisks might not contain the same data as the image mode VDisk prior to the deletion.
- If there are managed disks in the group, all disks are deleted from the group. They are returned to the unmanaged state.
- The group is deleted.

Attention: If you use the **-force** parameter to delete all the managed disk groups in your cluster, you are returned to the processing state where you were after you added nodes to the cluster. All data that is contained on the virtual disks is lost and cannot be recovered.

An invocation example

```
svctask rmmdiskgrp -force Group3
```

The resulting output

```
No feedback
```

Chapter 13. Managed disk commands

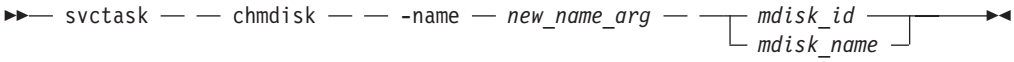
The following commands enable you to work with managed disk options with the SAN Volume Controller.

If the cluster detects an MDisk, it automatically adds it to the list of known MDisks. If you subsequently delete the RAID that corresponds to the MDisk, the cluster only deletes the MDisk from the list if the MDisk is offline and it has a mode of unmanaged (it does not belong to an MDisk group).

chmdisk

Use the **chmdisk** command to modify the name of a managed disk (MDisk).

Syntax



Parameters

-name *new_name_arg*
(Required) Specifies the new name to be applied to the managed disk.

mdisk_id | *mdisk_name*
(Required) Specifies the ID or name of the managed disk to modify.

Description

This command modifies the name, or label, that is assigned to a given managed disk. You can subsequently use the new name to refer to the managed disk.

An invocation example

```
svctask chmdisk -name testmdisk mdisk0
```

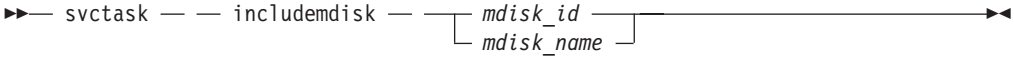
The resulting output

```
No feedback
```

includemdisk

Use the **includemdisk** command to include a disk that has been excluded by the cluster.

Syntax



Parameters

mdisk_id | *mdisk_name*
(Required) Specifies the ID or name of the managed disk to add back into the cluster.

Description

The specified managed disk is included in the cluster.

You might exclude a disk from the cluster because of multiple I/O failures. These failures might be caused by noisy links. Once a fabric-related problem has been fixed, the excluded disk can be added back into the cluster.

Running this command against an MDisk might change its state, whether the state is reported as excluded.

Note: If an MDisk is in the excluded state, is offline, and does not belong to an MDisk group, issuing an include command for this MDisk results in the MDisk record being deleted from the cluster.

An invocation example

```
svctask includemdisk mdisk5
```

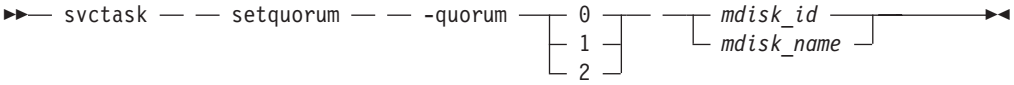
The resulting output

```
No feedback
```

setquorum

Use the **setquorum** command to change the managed disks (MDisks) that are assigned as quorum candidate disks.

Syntax



Parameters

-quorum 0 | 1 | 2
(Required) Specifies the quorum index.

mdisk_id | *mdisk_name*
(Required) Specifies the ID or name of the managed disk to assign as a quorum disk.

Description

This command sets the managed disk to the specified quorum index.

The cluster uses the quorum disk as a tie breaker when exactly half of the nodes that were previously a member of the cluster is present.

The use of a quorum disk allows the cluster to manage a SAN fault that splits the cluster exactly in half. One half of the cluster continues to operate and the other half stops until SAN connectivity is restored.

There is only *one* quorum disk; however, the cluster uses three disks as quorum candidate disks. The cluster selects the actual quorum disk from the pool of quorum candidate disks. The quorum candidate disks also hold a copy of important cluster metadata. Contiguous space in the amount of 256 MB is reserved for this purpose on each quorum candidate disk.

When you issue this command, the MDisk that currently is assigned the quorum index number is set to a nonquorum disk. The cluster automatically assigns quorum indexes.

An invocation example

```
svctask setquorum -quorum 2 mdisk7
```

The resulting output

```
No feedback
```

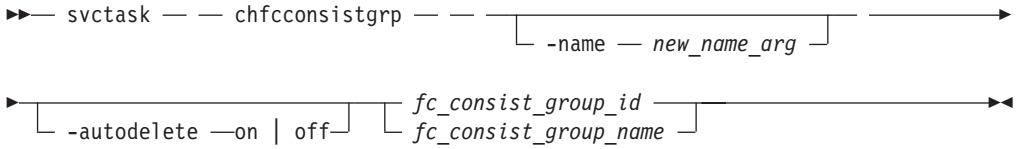
Chapter 14. FlashCopy commands

The following commands enable you to work with FlashCopy methods and functions with the SAN Volume Controller.

chfcconsistgrp

The **chfcconsistgrp** command changes the name of a consistency group or marks the group for auto-deletion.

Syntax



Parameters

-name *new_name_arg*
(Optional) Specifies the new name to assign to the consistency group.

-autodelete on | off
(Optional) Deletes the consistency group when the last mapping that it contains is deleted or removed from the consistency group.

fc_consist_group_id | fc_consist_group_name
(Required) Specifies the ID or existing name of the consistency group that you want to modify.

Description

The **chfcconsistgrp** command changes the name of a consistency group, marks the group for auto-deletion, or both.

An invocation example

```
svctask chfcconsistgrp -name testgrp1 fcconsistgrp1
```

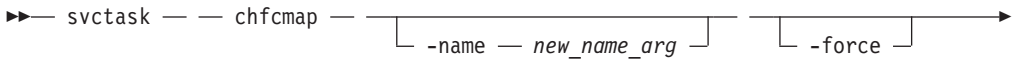
The resulting output

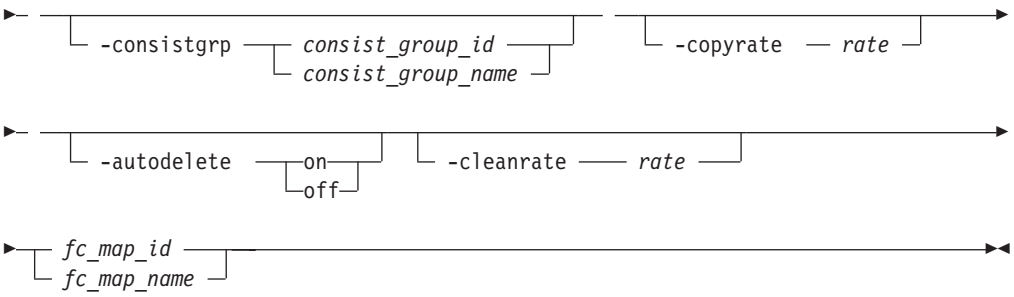
```
No feedback
```

chfcmap

The **chfcmap** command modifies attributes of an existing mapping.

Syntax





Parameters

-name *new_name_arg*
(Optional) Specifies the new name to assign to the mapping. The **-name** parameter cannot be used with any other optional parameters.

-force
(Optional) Specifies that the mapping be modified to a stand-alone mapping (equivalent to creating the mapping without a consistency group ID). You cannot specify the **-force** parameter with the **-consistgrp** parameter.

-consistgrp *consist_group_id | consist_group_name*
(Optional) Specifies the consistency group for which you want to modify the mapping. You cannot specify the **-consistgrp** parameter with the **-force** parameter.

Note: The consistency group cannot be modified if the specified consistency group is in the **preparing**, **prepared**, **copying**, **suspended**, or **stopping** state.

-copyrate *rate*
(Optional) Specifies the copy rate. The *rate* value can be **0** - **100**. The default value is **50**. A value of **0** indicates no background copy process. For the supported **-copyrate** values and their corresponding rates, see Table 7 on page 139.

-autodelete on | off
(Optional) Specifies that the autodelete function be turned on or off for the specified mapping. When you specify the **-autodelete on** parameter, you are deleting a mapping after the background copy completes. If the background copy is already complete, the mapping is deleted immediately.

-cleanrate *rate*
(Optional) Sets the cleaning rate for the mapping. The *rate* value can be **0** - **100**. The default value is **50**.

fc_map_id | fc_map_name
(Required) Specifies the ID or name of the mapping to modify. Enter the ID or name last on the command line.

Description

The **svctask chfcmap** command modifies attributes of an existing mapping.

Attention: You must enter the *fc_map_id | fc_map_name* last on the command line.

If you have created several FlashCopy mappings for a group of VDisks that contain elements of data for the same application, you can assign these mappings

to a single FlashCopy consistency group. You can then issue a single prepare command and a single start command for the whole group, for example, so that all of the files for a particular database are copied at the same time.

The **copyrate** parameter specifies the copy rate. If **0** is specified, background copy is disabled. The **cleanrate** parameter specifies the rate for cleaning the target VDisk. The cleaning process is only active if the mapping is in the **copying** state and the background copy has completed, the mapping is in the **copying** state and the background copy is disabled, or the mapping is in the **stopping** state. You can disable cleaning when the mapping is in the **copying** state by setting the **cleanrate** parameter to **0**. If the **cleanrate** is set to **0**, the cleaning process runs at the default rate of **50** when the mapping is in the **stopping** state to ensure that the stop operation completes.

Table 7 provides the relationship of the copy *rate* and cleaning *rate* values to the attempted number of grains to be split per second. A grain is the unit of data represented by a single bit.

Table 7. Relationship between the rate, data rate and grains per second values

User-specified *rate* **attribute value**	**Data copied/sec**	**256 KB grains/sec**	**64 KB grains/sec**
1 - 10	128 KB	0.5	2
11 - 20	256 KB	1	4
21 - 30	512 KB	2	8
31 - 40	1 MB	4	16
41 - 50	2 MB	8	32
51 - 60	4 MB	16	64
61 - 70	8 MB	32	128
71 - 80	16 MB	64	256
81 - 90	32 MB	128	512
91 - 100	64 MB	256	1024

An invocation example

```
svctask chfcmap -name testmap 1
```

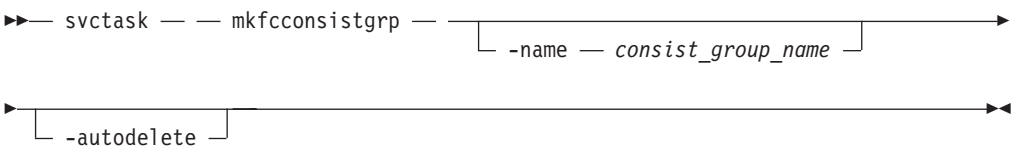
The resulting output

```
No feedback
```

mkfcconsistgrp

The **mkfcconsistgrp** command creates a new FlashCopy consistency group and identification name.

Syntax



Parameters

-name *consist_group_name*
(Optional) Specifies a name for the consistency group. If you do not specify a consistency group name, a name is automatically assigned to the consistency group. For example, if the next available consistency group ID is id=2, the consistency group name is fccstgrp2.

-autodelete
(Optional) Deletes the consistency group when the last mapping that it contains is deleted or removed from the consistency group.

Description

This command creates a new consistency group and identification name. The ID of the new group is displayed when the command process completes.

If you have created several FlashCopy mappings for a group of VDisks that contain elements of data for the same application, you might find it convenient to assign these mappings to a single FlashCopy consistency group. You can then issue a single prepare command and a single start command for the whole group, for example, so that all of the files for a particular database are copied at the same time.

An invocation example

```
svctask mkfcconsistgrp
```

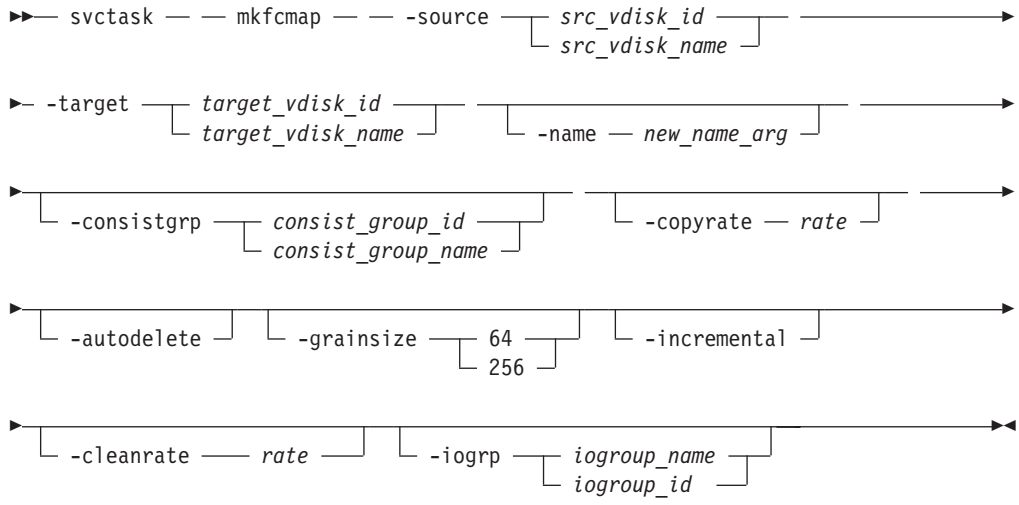
The resulting output

```
FlashCopy Consistency Group, id [1], successfully created
```

mkfcmap

The **mkfcmap** command creates a new FlashCopy mapping, which maps a source virtual disk to a target virtual disk for subsequent copying.

Syntax



Parameters

-source *src_vdisk_id | src_vdisk_name*
(Required) Specifies the ID or name of the source virtual disk.

-target *target_vdisk_id | target_vdisk_name*
(Required) Specifies the ID or name of the target virtual disk.

-name *new_name_arg*
(Optional) Specifies the name to assign to the new mapping.

-consistgrp *consist_group_id | consist_group_name*
(Optional) Specifies the consistency group to add the new mapping to. If you do not specify a consistency group, the mapping is treated as a stand-alone mapping.

-copyrate *rate*
(Optional) Specifies the copy rate. The *rate* value can be **0** - **100**. The default value is **50**. A value of **0** indicates no background copy process. For the supported **-copyrate** values and their corresponding rates, see Table 8 on page 142.

-autodelete
(Optional) Specifies that a mapping be deleted when the background copy completes. The default, which applies if this parameter is not entered, is that **autodelete** is set to off.

-grainsize 64 | 256
(Optional) Specifies the grain size for the mapping. The default value is **256**. Once set, this value cannot be changed.

-incremental
(Optional) Marks the FlashCopy mapping as an incremental copy. The default is nonincremental. Once set, this value cannot be changed.

-cleanrate *rate*
(Optional) Sets the cleaning rate for the mapping. The *rate* value can be **0** - **100**. The default value is **50**.

-iogrp *iogroup_name | iogroup_id*
(Optional) Specifies the I/O group for the FlashCopy bitmap. Once set, this value cannot be changed. The default I/O group is either the source VDisk, if a single target map, or the I/O group of the other FlashCopy mapping to which either the source or target VDisks belong.

Description

This command creates a new FlashCopy mapping. This mapping persists until it is manually deleted, or until it is automatically deleted when the background copy completes and the **autodelete** parameter set to **on**. The source and target VDisks must be specified on the **mkfcmap** command. The **mkfcmap** command fails if the source and target VDisks are not identical in size. Issue the **svcinfo lsvdisk -bytes** command to find the exact size of the source VDisk for which you want to create a target disk of the same size. The target VDisk that you specify cannot be a target VDisk in an existing FlashCopy mapping. A mapping cannot be created if the resulting set of connected mappings exceeds 256 connected mappings.

The mapping can optionally be given a name and assigned to a consistency group, which is a group of mappings that can be started with a single command. These are groups of mappings that can be processed at the same time. This enables multiple virtual disks to be copied at the same time, which creates a consistent

copy of multiple disks. This consistent copy of multiple disks is required by some database products in which the database and log files reside on different disks.

The **copyrate** parameter specifies the copy rate. If **0** is specified, background copy is disabled. The **cleanrate** parameter specifies the rate for cleaning the target VDisk. The cleaning process is only active if the mapping is in the **copying** state and the background copy has completed, the mapping is in the **copying** state and the background copy is disabled, or the mapping is in the **stopping** state. You can disable cleaning when the mapping is in the **copying** state by setting the **cleanrate** parameter to **0**. If the **cleanrate** is set to **0**, the cleaning process runs at the default rate of **50** when the mapping is in the **stopping** state to ensure that the stop operation completes.

Table 8 provides the relationship of the copy *rate* and cleaning *rate* values to the attempted number of grains to be split per second. A grain is the unit of data represented by a single bit.

Table 8. Relationship between the rate, data rate and grains per second values

User-specified *rate* **attribute value**	**Data copied/sec**	**256 KB grains/sec**	**64 KB grains/sec**
1 - 10	128 KB	0.5	2
11 - 20	256 KB	1	4
21 - 30	512 KB	2	8
31 - 40	1 MB	4	16
41 - 50	2 MB	8	32
51 - 60	4 MB	16	64
61 - 70	8 MB	32	128
71 - 80	16 MB	64	256
81 - 90	32 MB	128	512
91 - 100	64 MB	256	1024

An invocation example

```
svctask mkfcmap -source 0 -target 2 -name mapone
```

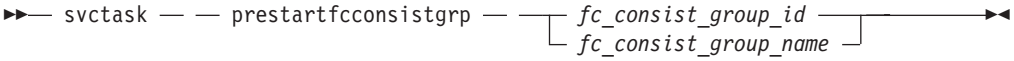
The resulting output

```
FlashCopy Mapping, id [1], successfully created
```

prestartfcconsistgrp

The **prestartfcconsistgrp** command prepares a consistency group (a group of FlashCopy mappings) so that the consistency group can be started. This command flushes the cache of any data that is destined for the source virtual disk and forces the cache into the write-through mode until the consistency group is started.

Syntax



Parameters

fc_consist_group_id | fc_consist_group_name
(Required) Specifies the name or ID of the consistency group that you want to prepare.

Description

This command prepares a consistency group (a group of FlashCopy mappings) to subsequently start. The preparation step ensures that any data that resides in the cache for the source virtual disk is first flushed to disk. This step ensures that the FlashCopy target VDisk is identical to what has been acknowledged to the host operating system as having been written successfully to the source VDisk.

You must issue the **svctask prestartfcconsistgrp** command to prepare the FlashCopy consistency group before the copy process can be started. When you have assigned several mappings to a FlashCopy consistency group, you must issue a single prepare command for the whole group to prepare all of the mappings at once.

The consistency group must be in the **idle_or_copied** or **stopped** state before it can be prepared. When you enter the **prestartfcconsistgrp** command, the group enters the preparing state. After the preparation is complete, the consistency group status changes to **prepared**. At this point, you can start the group.

If FlashCopy mappings are assigned to a consistency group, the preparing and the subsequent starting of the mappings in the group must be performed on the consistency group rather than on an individual FlashCopy mapping that is assigned to the group. Only stand-alone mappings, which are mappings that are not assigned to a consistency group, can be prepared and started on their own. A FlashCopy mapping must be prepared before it can be started.

An invocation example

```
svctask prestartfcconsistgrp 1
```

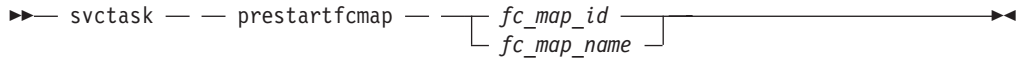
The resulting output

```
No feedback
```

prestartfcmap

The **prestartfcmap** command prepares a FlashCopy mapping so that it can be started. This command flushes the cache of any data that is destined for the source virtual disk and forces the cache into the write-through mode until the mapping is started.

Syntax



Parameters

fc_map_id | fc_map_name
(Required) Specifies the name or ID of the mapping to prepare.

Description

This command prepares a single mapping for subsequent starting. The preparation step ensures that any data that resides in the cache for the source virtual disk is first transferred to disk. This step ensures that the copy that is made is consistent with what the operating system expects on the disk.

Note: To prepare a FlashCopy mapping that is part of a consistency group, you must use the **prestartfcconsistgrp** command.

The mapping must be in the **idle_or_copied** or **stopped** state before it can be prepared. When the **prestartfcmap** command is processed, the mapping enters the **preparing** state. After the preparation is complete, it changes to the prepared state. At this point, the mapping is ready to start.

Attention: This command can take a considerable amount of time to complete.

An invocation example

```
svctask prestartfcmap 1
```

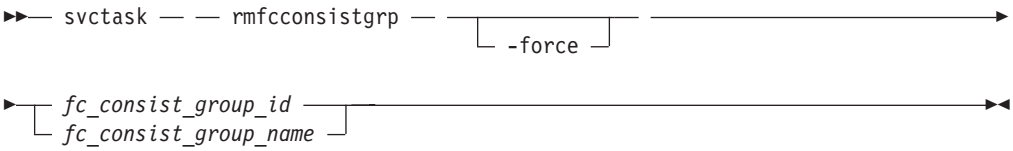
The resulting output

```
No feedback
```

rmfcconsistgrp

The **rmfcconsistgrp** command deletes a FlashCopy consistency group.

Syntax



Parameters

-force
(Optional) Specifies that all of the mappings that are associated with a consistency group that you want to delete are removed from the group and changed to stand-alone mappings. This parameter is only required if the consistency group that you want to delete contains mappings.

fc_consist_group_id | *fc_consist_group_name*
(Required) Specifies the ID or name of the consistency group that you want to delete.

Description

This command deletes the specified FlashCopy consistency group. If there are mappings that are members of the consistency group, the command fails unless you specify the **-force** parameter. When you specify the **-force** parameter, all of the mappings that are associated with the consistency group are removed from the group and changed to stand-alone mappings.

To delete a single mapping in the consistency group, you must use the **svctask rmfcmap** command.

An invocation example

```
svctask rmfcconsistgrp fcconsistgrp1
```

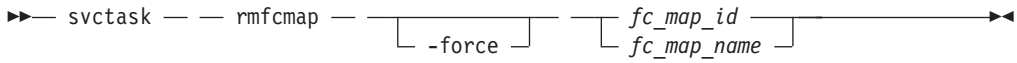
The resulting output

```
No feedback
```

rmfcmap

The **rmfcmap** command deletes an existing mapping.

Syntax



Parameters

-force
(Optional) Specifies that the target VDisk is brought online. This parameter is required if the FlashCopy mapping is in the stopped state.

fc_map_id | fc_map_name
(Required) Specifies the ID or name of the FlashCopy mapping to delete. Enter the ID or name last on the command line.

Description

The **rmfcmap** command deletes the specified mapping if the mapping is in the **idle_or_copied** or **stopped** state. If it is in the stopped state, the **-force** parameter is required. If the mapping is in any other state, you must stop the mapping before you can delete it.

Deleting a mapping only deletes the logical relationship between the two virtual disks; it does not affect the virtual disks themselves. However, if you force the deletion, the target virtual disk (which might contain inconsistent data) is brought back online.

An invocation example

```
svctask rmfcmap testmap
```

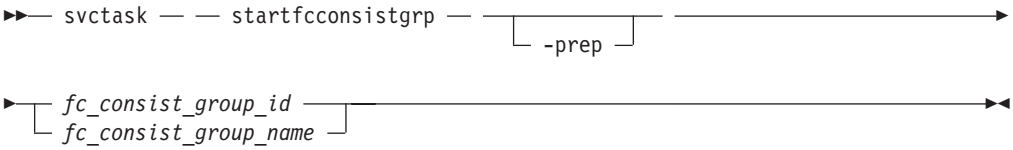
The resulting output

```
No feedback
```

startfcconsistgrp

The **startfcconsistgrp** command starts a FlashCopy consistency group of mappings. This command makes a point-in-time copy of the source virtual disks at the moment that the command is started.

Syntax



Parameters

-prep
(Optional) Specifies that the designated FlashCopy consistency group be prepared prior to starting the FlashCopy consistency group. A FlashCopy consistency group must be prepared before it can be started. When you use this parameter, the system automatically issues the **prestartfcconsistgrp** command for the group that you specify.

fc_consist_group_id | *fc_consist_group_name*
(Required) Specifies the ID or name of the consistency group mapping to start.

Description

This command starts a consistency group mapping, which results in a point-in-time copy of the source virtual disks.

Note: The **startfcconsistgrp** command can take some time to process particularly if you have specified the **-prep** parameter. If you use the **-prep** parameter, you give additional processing control to the system because the system must prepare the mapping before the mapping is started. If the prepare process takes too long, the system completes the prepare but does not start the consistency group. In this case, error message CMMVC6209E displays. To control the processing times of the **prestartconsistgrp** and **startfcconsistgrp** commands independently of each other, do not use the **-prep** parameter. Instead, first issue the **prestartfcconsistgrp** command, and then to start the copy, issue the **startfcconsistgrp** command.

An invocation example

```
svctask startfcconsistgrp -prep 2
```

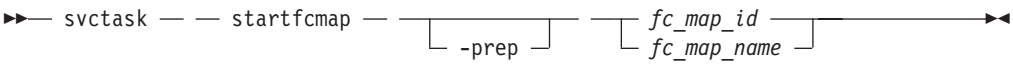
The resulting output

```
No feedback
```

startfcmap

The **startfcmap** command starts a FlashCopy mapping. This command makes a point-in-time copy of the source virtual disk at the moment that the command is started.

Syntax



Parameters

-prep
(Optional) Specifies that the designated mapping be prepared prior to starting the mapping. A mapping must be prepared before it can be started. When you use this parameter, the system automatically issues the **prestartfcmap** command for the group that you specify.

fc_map_id | fc_map_name
(Required) Specifies the ID or name of the mapping to start.

Description

This command starts a single mapping, which results in a point-in-time copy of the source virtual disk.

Note: The **startfcmap** command can take some time to start, particularly if you use the **-prep** parameter. If you use the **-prep** parameter, you give additional starting control to the system. The system must prepare the mapping before the mapping is started. To keep control when the mapping starts, you must issue the **prestartfcmap** command before you issue the **startfcmap** command.

An invocation example

```
svctask startfcmap -prep 2
```

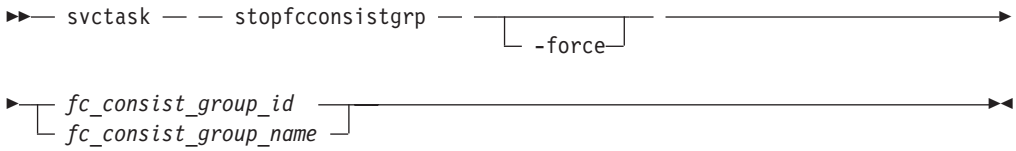
The resulting output

```
No feedback
```

stopfcconsistgrp

The **stopfcconsistgrp** command stops all processing that is associated with a FlashCopy consistency group that is in one of the following processing states: **prepared**, **copying**, **stopping**, or **suspended**.

Syntax



Parameters

-force
(Optional) Specifies that all processing that is associated with the mappings of the designated consistency group be stopped immediately.

Note: When you use this parameter, all FlashCopy mappings that depend on the mappings in this group (as listed by the **lsfcmapdependentmaps** command) are also stopped.

fc_consist_group_id | fc_consist_group_name
(Required) Specifies the name or ID of the consistency group that you want to stop.

Description

This command stops a group of mappings in a consistency group. If the copy process is stopped, the target disks become unusable unless they already contain complete images of the source. Disks that contain complete images of the source have a progress of 100, as indicated in the **-lsfcmap** command output. The target VDisk is reported as offline if it does not contain a complete image. Before you can access this VDisk, the group of mappings must be prepared and restarted.

If the consistency group is in the **idle_or_copied** state, then the **stopfcconsistgrp** command has no effect and the consistency group stays in the **idle_or_copied** state.

Note: Prior to SVC 4.2.0, the **stopfcconsistgrp** command always caused the consistency group to go to the **stopped** state, taking the target VDisks offline.

An invocation example

```
svctask stopfcconsistgrp testmapone
```

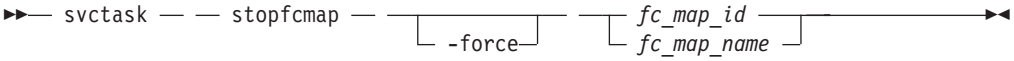
The resulting output

```
No feedback
```

stopfcmap

The **stopfcmap** command stops all processing that is associated with a FlashCopy mapping that is in one of the following processing states: **prepared**, **copying**, **stopping**, or **suspended**.

Syntax



Parameters

-force
(Optional) Specifies that all processing that is associated with the designated mapping be stopped immediately.

Note: When you use this parameter, all FlashCopy mappings that depend on this mapping (as listed by the **-lsfcmapdependentmaps** command) are also stopped.

fc_map_id | *fc_map_name*
(Required) Specifies the name or ID of the mapping to stop.

Description

This command stops a single mapping. If the copy process is stopped, the target disk becomes unusable unless it already contained a complete image of the source (that is, unless the map had a progress of 100 as shown by the **-lsfcmap** command). Before you can use the target disk, the mapping must once again be prepared and then reprocessed (unless the target disk already contained a complete image).

Only stand-alone mappings can be stopped using the **stopfcmap** command. Mappings that belong to a consistency group must be stopped using the **stopfcconsistgrp** command.

If the mapping is in the **idle_or_copied** state, the **stopfcmap** command has no effect and the mapping stays in the **idle_or_copied** state.

Note: Before SAN Volume Controller 4.2.0, the **stopfcmap** command always changed the mapping state to **stopped** and took the target VDisk offline. This change can break scripts that depend on the previous behavior.

An invocation example

```
svctask stopfcmap testmapone
```

The resulting output

```
No feedback
```

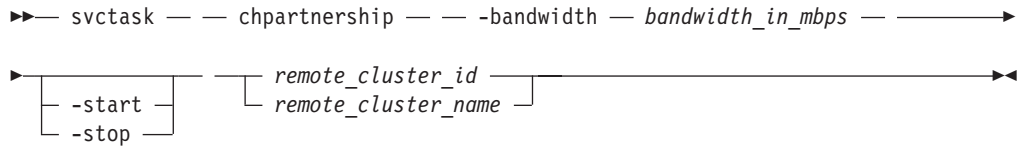
Chapter 15. Metro Mirror and Global Mirror commands

The following Copy Service commands enable you to work with the Metro Mirror and Global Mirror services that the SAN Volume Controller provides.

chpartnership

The **chpartnership** command modifies the bandwidth of the partnership between the local cluster and the remote cluster that is specified in the command. This affects the bandwidth that is available for background copy in a cluster partnership by either Metro Mirror or Global Mirror operations. This command can also be used to disable and re-enable the partnership, to permit the local cluster to be disconnected and then reconnected to the remote cluster.

Syntax



Parameters

-bandwidth *bandwidth_in_mbps*
(Required) Specifies the new bandwidth in megabytes per second (MBps). This parameter might be set to a value that is greater than the intercluster links can sustain. If so, the actual copy rate defaults to what is available on the link. The default bandwidth is 50.

-start | -stop
(Optional) Starts or stops a Metro Mirror or Global Mirror partnership. To start or stop a partnership, run the **svctask chpartnership** command from either cluster.

remote_cluster_id | remote_cluster_name
(Required) Specifies the cluster ID or name of the remote cluster. The intracluster bandwidth cannot be modified, so if you enter the local cluster name or ID, an error occurs.

Description

This command modifies the bandwidth of the partnership between the local cluster and the remote cluster that is specified in the command. This affects the bandwidth that is available for a background copy in Metro Mirror or Global Mirror relationships, in the direction from the local to the remote cluster. To modify the background copy bandwidth in the other direction (remote cluster –> local cluster), it is necessary to issue the corresponding **chpartnership** command to the remote cluster.

When you stop the cluster partnership, you temporarily disable the partnership and disconnect the local cluster from the remote cluster. The configuration is retained. The cluster partnership must be in either the

partially_configured_stopped or fully_configured_stopped states to be started.

An invocation example

```
svctask chpartnership -bandwidth 20 cluster1
svctask chpartnership -stop cluster1
```

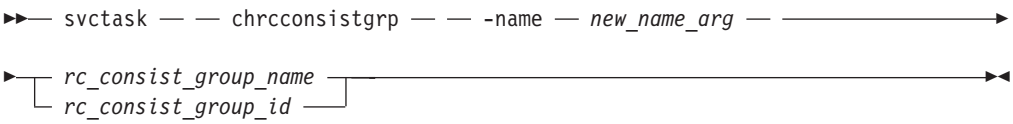
The resulting output

```
No feedback
```

chrcconsistgrp

The **chrcconsistgrp** command modifies the name of an existing Metro Mirror or Global Mirror consistency group.

Syntax



Parameters

-name *new_name_arg*
(Required) Specifies the new name to assign to the consistency group.

rc_consist_group_name | *rc_consist_group_id*
(Required) Specifies the ID or existing name of the consistency group that you want to modify.

Description

This command changes the name of the specified consistency group.

An invocation example

Change the name of the consistency group called `rc_testgrp` to `rctestone`.

```
svctask chrcconsistgrp -name rctestone rc_testgrp
```

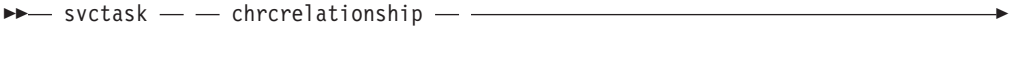
The resulting output

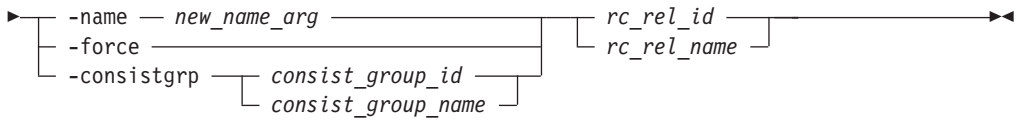
```
No feedback
```

chrcrelationship

The **chrcrelationship** command enables you to modify certain attributes of an existing relationship, such as to add a relationship to a consistency group, to remove a relationship from a consistency group, and to change the name of the relationship. You can only change one attribute at a time per command submittal.

Syntax





Parameters

-name *new_name_arg*
(Optional) Specifies a new label to assign to the relationship.

This parameter is required if you do not specify the **-consistgrp** or **-force** parameter.

-consistgrp *consist_group_id* | *consist_group_name*
(Optional) Specifies a new consistency group to assign the relationship to. Only relationships of the same copy type (Metro Mirror or Global Mirror) can be assigned to the same consistency group. You cannot use this parameter with the **-name**, or **-force** parameters.

This parameter is required if you do not specify the **-name** or **-force** parameter.

-force
(Optional) Specifies that you want the system to remove the relationship from a consistency group making the relationship a stand-alone relationship. You cannot use this parameter with the **-name** or **-consistgrp** parameters.

This parameter is required if you do not specify the **-name** or **-consistgrp** parameter.

rc_rel_name | *rc_rel_id*
(Required) Specifies the ID or name of the relationship.

Description

This command modifies the specified attributes of the supplied relationship, one attribute at a time; you cannot use more than one of the three optional parameters in the same command. In addition to changing the name of a consistency group, this command can be used for the following purposes.

- You can add a stand-alone relationship to a consistency group by specifying the **-consistgrp** parameter and the name or ID of the consistency group. The relationship and consistency group must be connected when the command is issued and must share the following components:
  - Master cluster
  - Auxiliary cluster
  - State (unless the group is empty)
  - Primary (unless the group is empty)
  - Type (unless the group is empty)

  When the first relationship is added to an empty group, the group takes on the same state, primary (copy direction), and type (Metro or Global Mirror) as the relationship. Subsequent relationships must have the same state, copy direction, and type as the group in order to be added to it. A relationship can only belong to one consistency group.
- You can remove a relationship from a consistency group by specifying the **-force** parameter and the name or ID of the relationship. Although you do not have to specify or confirm the name of the consistency group, verify which group the relationship belongs to before you issue this command.

This form of the modify relationship command succeeds in the connected or disconnected states. If the clusters are disconnected the relationship is only removed from the consistency group on the local cluster, at the time the command is issued. When the clusters are reconnected the relationship is automatically removed from the consistency group on the other cluster. Alternatively, you can issue an explicit modify (**chrcrelationship**) command to remove the relationship from the group on the other cluster while it is still disconnected.

Note: If you remove all relationships from the group, the relationship type is reset to **empty_group**. When you add a relationship to the empty group, the group again takes on the same type as the relationship.

- To move a relationship between two consistency groups, you must issue the **chrcrelationship** command twice. Use the **-force** parameter to remove the relationship from its current group, and then use the **-consistgrp** parameter with the name of the new consistency group.

An invocation example

Change the name of the relationship `rccopy1` to `testrel`

```
svctask chrcrelationship -name testrel rccopy1
```

Add relationship `rccopy2` to group called `newgroup`.

```
svctask chrcrelationship -consistgrp newgroup rccopy2
```

Remove relationship `rccopy3` from whichever consistency group it is a member of.

```
svctask chrcrelationship -force rccopy3
```

The resulting output

```
No feedback
```

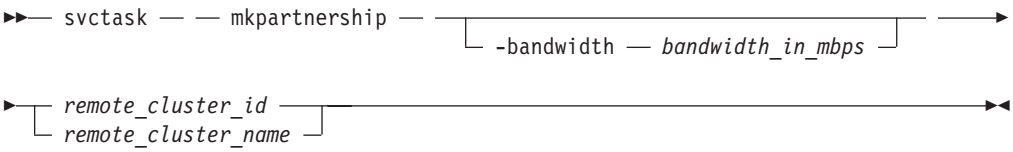
There is no feedback in any of these cases.

mkpartnership

The **mkpartnership** command establishes a one-way Metro Mirror or Global Mirror relationship between the local cluster and a remote cluster.

Syntax

To establish a fully functional Metro Mirror or Global Mirror partnership, you must issue this command to both clusters. This step is a prerequisite to creating Metro Mirror or Global Mirror relationships between VDisks on the clusters.



Parameters

-bandwidth *bandwidth_in_mbps*
(Optional) Specifies the bandwidth that is used by the background copy process between the clusters. It adjusts the bandwidth that is used by Metro

Mirror or Global Mirror for the initial background copy process. The bandwidth defaults to 50 MBps (megabytes per second) if you do not specify it. Set the bandwidth to a value that is less than or equal to the bandwidth that can be sustained by the intercluster link. If the **-bandwidth** parameter is set to a higher value than the link can sustain, the background copy process uses the actual available bandwidth. See the *IBM System Storage SAN Volume Controller Software Installation and Configuration Guide* for more information about the effect that background copy bandwidth has on foreground I/O latency.

remote_cluster_id | remote_cluster_name
(Required) Specifies the cluster ID or name of the remote cluster. Issue the **svcinfo lsclustercandidate** command to list the remote clusters that are available. If two or more remote clusters have the same name and the name is included in this command, the command fails and it requests the ID of the cluster instead of the name.

Description

This command creates a one-way partnership between the local cluster and the remote cluster that you specify in the command. To create a two-way partnership, the equivalent **svctask mkpartnership** command must be issued from the other cluster.

Intercluster Mirror relationships can be created between primary VDisks in the local cluster and auxiliary VDisks in the remote cluster. Intracluster relationships can be created between VDisks that reside in a local cluster. The VDisks must belong to the same I/O group within the cluster.

An invocation example

```
svctask mkpartnership -bandwidth 20 cluster1
```

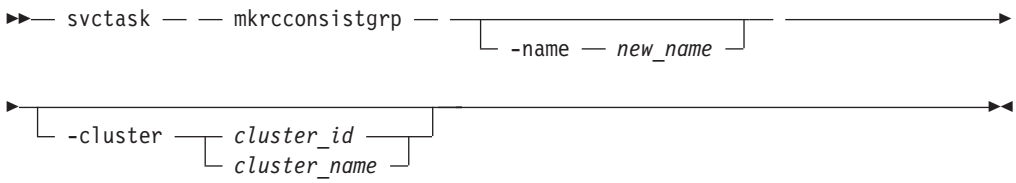
The resulting output

```
No feedback
```

mkrcconsistgrp

The **mkrcconsistgrp** command creates a new, empty Metro Mirror or Global Mirror consistency group. If the **-cluster** parameter is not specified, the consistency group is created on the local cluster only.

Syntax



Parameters

-name *new_name*
(Optional) Specifies a name for the new consistency group.

-cluster *cluster_id* | *cluster_name*
(Optional) Specifies the name or ID of the remote cluster. If **-cluster** is not specified, a consistency group is created only on the local cluster.

Description

This command creates a new consistency group. The ID of the new group is displayed after the command processes. The name must be unique across all consistency groups that are known to the clusters within this consistency group. If the consistency group involves two clusters, the clusters must be in communication throughout the create process.

The new consistency group does not contain any relationships and will be in the empty state. You can add Metro Mirror or Global Mirror relationships to the group using the **svctask chrelationship** command.

An invocation example

```
svctask mkrcconsistgrp -name rc_testgrp
```

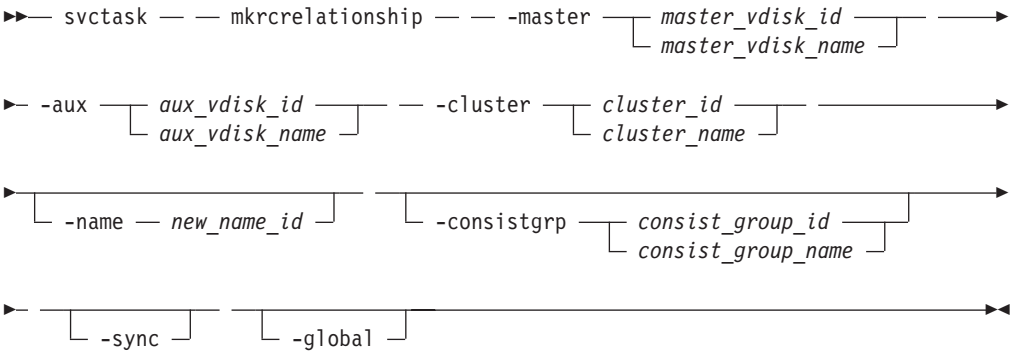
The resulting output

```
RC Consistency Group, id [255], successfully created
```

mkrcrelationship

The **mkrcrelationship** command creates a new Metro Mirror or Global Mirror relationship with virtual disks (VDisks) in the same cluster (intracluster relationship) or in two different clusters (intercluster relationship).

Syntax



Parameters

-master *master_vdisk_id* | *master_vdisk_name*
(Required) Specifies the ID or name of the master virtual disk.

-aux *aux_vdisk_id* | *aux_vdisk_name*
(Required) Specifies the ID or name of the auxiliary virtual disk.

-cluster *cluster_id* | *cluster_name*
(Required) Specifies the ID or name of the remote cluster.

If you are creating an intracluster relationship, enter the ID of the local cluster. The VDisks in the relationship must belong to the same I/O group within the cluster.

If you are creating an intercluster relationship, enter the ID of the remote cluster. To create a relationship in two different clusters, the clusters must be connected at the time that the **svctask mkrcrelationship** command is received.

-name *new_name_id*
(Optional) Specifies a label to assign to the relationship.

-consistgrp *consist_group_id | consist_group_name*
(Optional) Specifies a consistency group that this relationship joins. If you do not supply the **-consistgrp** parameter, the relationship is created as a stand-alone relationship that can be started, stopped, and switched on its own.

Note: Metro and Global Mirror relationships cannot belong to the same consistency group. When the first relationship is added to the consistency group, the group takes on the same type as the relationship. Subsequently, only relationships of that type can be added to the consistency group.

-sync
(Optional) Specifies that you want the system to create a synchronized relationship. The **-sync** parameter guarantees that the master and auxiliary disks contain identical data at the point that the relationship is created. You must ensure that the auxiliary disk is created to match the master disk and that no input transactions take place to either disk before you issue the create command. The initial background synchronization is skipped.

-global
(Optional) Specifies that you want the system to create a new Global Mirror relationship. If you do not specify the **-global** parameter, a Metro Mirror relationship is created instead.

Description

This command creates a new Metro Mirror or Global Mirror relationship. A Metro Mirror relationship defines the relationship between two virtual disks (VDisks): a master VDisk and an auxiliary VDisk. This relationship persists until it is deleted. The auxiliary virtual disk must be identical in size to the master virtual disk or the command fails, and if both VDisks are in the same cluster, they must both be in the same I/O group. The master and auxiliary cannot be in an existing relationship. Neither disk can be the target of a FlashCopy mapping. The command also returns the new relationship ID.

Metro Mirror relationships use one of the following copy types:

- A Metro Mirror copy ensures that updates are committed to both the primary and secondary VDisks before sending confirmation of I/O completion to the host application. This ensures that the secondary VDisk is synchronized with the primary VDisk in the event that a failover operation is performed.
- A Global Mirror copy allows the host application to receive confirmation of I/O completion before the updates are committed to the secondary VDisk. If a failover operation is performed, the host application must recover and apply any updates that were not committed to the secondary VDisk.

You can optionally give the relationship a name. The name must be a unique relationship name across both clusters.

The relationship can optionally be assigned to a consistency group. A consistency group ensures that a number of relationships are managed so that, in the event of

a disconnection of the relationships, the data in all relationships within the group is in a consistent state. This can be important in, for example, a database application where data files and log files are stored on separate VDisks and consequently are managed by separate relationships. In the event of a disaster, the primary and secondary sites might become disconnected. As the disconnection occurs and the relationships stop copying data from the primary to the secondary site, there is no assurance that updates to the two separate secondary VDisks will stop in a consistent manner if the relationships that are associated with the VDisks are not in a consistency group.

For proper database operation, it is important that updates to the log files and the database data are made in a consistent and orderly fashion. It is crucial in this example that the logfile VDisk and the data VDisk at the secondary site are in a consistent state. This can be achieved by putting the relationships that are associated with these VDdisks into a consistency group. Both Metro Mirror and Global Mirror processing ensure that updates to both VDisks at the secondary site are stopped, leaving a consistent image based on the updates that occurred at the primary site.

If you specify a consistency group, both the group and the relationship must have been created using the same master cluster and the same auxiliary cluster. The relationship must not be a part of another consistency group. If the consistency group is *empty*, it acquires the type of the first relationship that is added to it. Therefore, each subsequent relationship that you add to the consistency group must have the same type.

If the consistency group is *not empty*, the consistency group and the relationship must be in the same state. If the consistency group is *empty*, it acquires the state of the first relationship that is added to it. If the state has an assigned copy direction, the direction of the consistency group and the relationship must match that direction.

If you do not specify a consistency group, a stand-alone relationship is created.

If you specify the **-sync** parameter, the master and auxiliary virtual disks contain identical data at the point when the relationship is created. You must ensure that the auxiliary is created to match the master and that no data movement occurs to either virtual disk before you issue the **svctask mkrcrelationship** command.

If you specify the **-global** parameter, a Global Mirror relationship is created. Otherwise, a Metro Mirror relationship is created instead.

An invocation example

```
svctask mkrcrelationship -master vdisk1 -aux vdisk2 -name rccopy1
 -cluster 0000020063432AFD
```

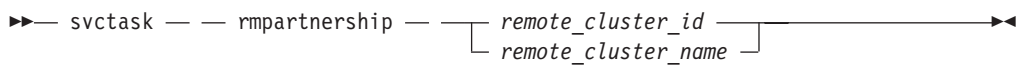
The resulting output

```
RC Relationship, id [28], successfully created
```

rmpartnership

The **rmpartnership** command removes the Metro Mirror or Global Mirror relationship on one cluster. Because the partnership exists on both clusters, it is necessary to run this command on both clusters to remove both sides of the partnership. If the command is run on only one cluster, the partnership enters a partially configured state on the other cluster.

Syntax



Parameters

remote_cluster_id | remote_cluster_name
(Required) Specifies the cluster ID or the name of the remote cluster.

Description

This command deletes one half of a partnership on a cluster. To remove the entire partnership, you must run the command twice, once on each cluster.

Attention: Before running the **svctask rmpartnership** command, you must remove all relationships and groups that are defined between the two clusters. To display cluster relationships and groups, run the **svcinfo lsrcrelationship** and **svcinfo lsrcconsistgrp** commands. To remove the relationships and groups that are defined between the two clusters, run the **svctask rmrcrelationship** and **svctask rmrcconsistgrp** commands.

An invocation example

```
svctask rmpartnership cluster1
```

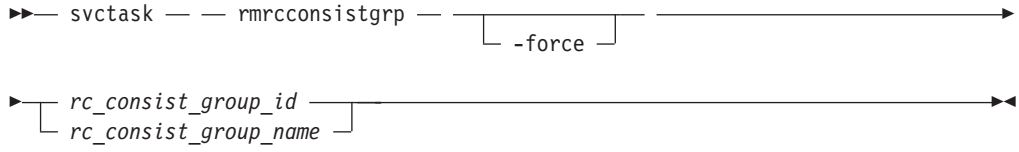
The resulting output

```
No feedback
```

rmrcconsistgrp

The **rmrcconsistgrp** command deletes an existing Metro Mirror or Global Mirror consistency group.

Syntax



Parameters

-force
(Optional) Specifies that you want the system to remove all relationships

belonging to a group before the consistency group is deleted. The relationships themselves are not deleted; they become stand-alone relationships.

Note: The **-force** parameter must be used to delete a consistency group when the consistency group has any Metro Mirror or Global Mirror relationships that is associated with it. If you do not use the **-force** parameter, the command fails.

rc_consist_group_id | rc_consist_group_name
(Required) Specifies the ID or the name of the consistency group to delete.

Description

This command deletes the specified consistency group. You can issue this command for any existing consistency group. If the consistency group is disconnected at the time that the command is issued, the consistency group is only deleted on the cluster that is connected. When the clusters reconnect, the consistency group is automatically deleted on the other cluster. Alternatively, if the clusters are disconnected, and you still want to remove the consistency group on both clusters, you can issue the **svctask rmrcconsistgrp** command separately on both of the clusters.

If the consistency group is not empty, the **-force** parameter is required to delete the group. This removes the relationships from the consistency group before the group is deleted. These relationships become stand-alone relationships. The state of these relationships is not changed by the action of removing them from the consistency group.

An invocation example

```
svctask rmrcconsistgrp rctestone
```

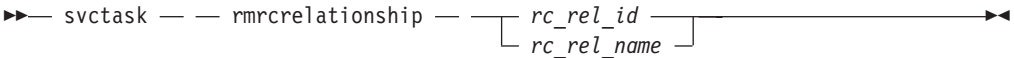
The resulting output

```
No feedback
```

rmrcrelationship

The **rmrcrelationship** command deletes an existing Metro Mirror or Global Mirror relationship.

Syntax



Parameters

rc_rel_id | rc_rel_name
(Required) Specifies the ID or the name of the relationship. A relationship cannot be deleted if it is part of a consistency group.

Description

This command deletes the relationship that is specified.

Deleting a relationship only deletes the logical relationship between the two virtual disks; it does not affect the virtual disks themselves.

If the relationship is disconnected at the time that the command is issued, the relationship is only deleted on the cluster where the command is being run. When the clusters reconnect, the relationship is automatically deleted on the other cluster. Alternatively, if the clusters are disconnected and if you still want to remove the relationship on both clusters, you can issue the **svctask rmrcrelationship** command independently on both of the clusters.

A relationship cannot be deleted if it is part of a consistency group. You must first remove the relationship from the consistency group using the **svctask chrcrelationship -force** command.

If you delete an inconsistent relationship, the secondary virtual disk becomes accessible even though it is still inconsistent. This is the one case in which Metro or Global Mirror does not inhibit access to inconsistent data.

An invocation example

```
svctask rmrcrelationship rccopy1
```

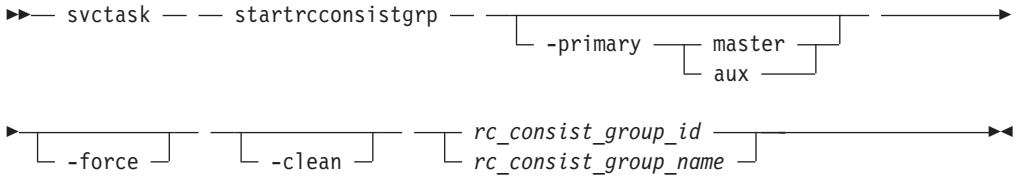
The resulting output

```
No feedback
```

startrcconsistgrp

The **startrcconsistgrp** command starts the Metro Mirror or Global Mirror consistency group copy process, sets the direction of copy if it is undefined, and optionally marks the secondary VDisks of the consistency group as clean.

Syntax



Parameters

-primary *master | aux*
(Optional) Specifies the copy direction by defining whether the master or auxiliary disk becomes the primary (source). This parameter is required when the primary is undefined if, for example, the consistency group is in the **Idling** state.

-force
(Optional) Specifies that you want the system to process the copy operation even if it might lead to a temporary loss of consistency while synchronization occurs. This parameter is required if the consistency group is in the **ConsistentStopped** state, but is not synchronized or is in the **Idling** state, but is not synchronized.

-clean
(Optional) Specifies that the VDisk that is to become a secondary is clean for each of the relationships belonging to the group; any changes made on the secondary VDisk are ignored, and only changes made on the clean primary

VDisk are considered during synchronization of the primary and secondary disks. The consistency group must be in an **Idling** (connected) state for this parameter to work.

rc_consist_group_id | rc_consist_group_name
(Required) Specifies the ID or name of the consistency group to start.

Description

This command starts a Metro Mirror or Global Mirror stand-alone consistency group.

This command can only be issued to a consistency group that is connected. For a consistency group that is idling, this command assigns a copy direction (primary and secondary roles) and begins the copy process. Otherwise, this command restarts a previous copy process that was stopped either by a stop command or by an I/O error.

If the resumption of the copy process leads to a period of time when the relationship is not consistent, then you must specify the **-force** parameter when you restart the relationship. This situation can arise if the relationship had been stopped and then further input transactions had been performed on the original primary disk of the relationship. When you use the **-force** parameter in this situation, the data on the secondary disk is not usable (because it is inconsistent) in a disaster recovery circumstance.

In the idling state, you must provide the **-primary** parameter. In other connected states, you can provide the **-primary** parameter, but it must match the existing setting.

The **-force** parameter is required if consistency would be lost by starting a copy operation. This can occur if write operations on either primary or secondary VDisks have taken place since the **ConsistentStopped** or **idling** state occurred. If the command is issued without the **-force** parameter in such circumstances, the command fails. In general, the **-force** parameter is required if the group is in one of the following states:

- Consistent_Stopped but not synchronized (sync=out_of_sync)
- Idling but not synchronized

The **-force** parameter is not required if the group is in one of the following states:

- Inconsistent_Stopped
- Inconsistent_Copying
- Consistent_Synchronized

However, the command does not fail if you specify the **-force** parameter.

The **-clean** parameter is used when a Metro Mirror or Global Mirror group is started and the secondary VDisks in this group are assumed to be clean. Clean in this sense, means that any changes that have been made at the secondary are ignored and only changes made at the primary are considered when synchronizing the primary and secondary VDisks. The **-clean** parameter can be used in the following scenario:

1. A consistency group is created with the **-synch** parameter. At this point, it does not matter if the primary and secondary contain the same data, even though the use of the **-synch** parameter implies that this is true.

2. A **stoprcconsistgrp** command is issued with the **-access** parameter. This permits access to the secondary disk. Change recording begins at the primary.
3. An image of the primary disk is copied and loaded on to the secondary disk. It is permissible to allow updates to the primary disk during the image copy as this image can be only a fuzzy image of the primary disk.
4. A **startrcconsistgrp** command that specifies the **-primary master**, **-force**, and **-clean** parameters is issued. The auxiliary disk is marked as clean and changes on the master disk that have occurred since the relationship was stopped are copied to the auxiliary disk.
5. Once the background copy has completed, relationships in the group become consistent and synchronized.

An invocation example

```
svctask startrcconsistgrp rccopy1
```

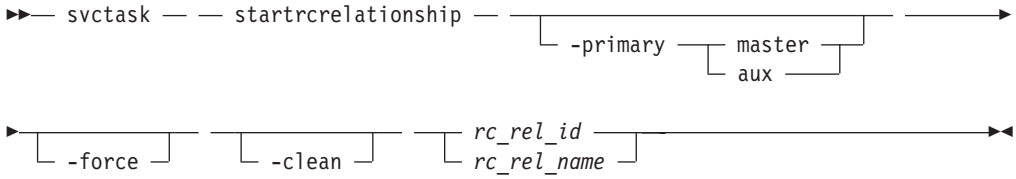
The resulting output

```
No feedback
```

startrcrelationship

The **startrcrelationship** command starts the Metro Mirror or Global Mirror relationship copy process, sets the direction of copy if undefined, and optionally, marks the secondary VDisk of the relationship as clean. The relationship must be a stand-alone relationship.

Syntax



Parameters

-primary *master* | *aux*
(Optional) Specifies the copy direction by defining whether the master or auxiliary disk becomes the primary (source). This parameter is required when the primary is undefined if, for example, the relationship is in the idling state.

-force
(Optional) Specifies that you want the system to process the copy operation even if it might lead to a temporary loss of consistency while synchronization occurs. This parameter is required if the relationship is in the **Consistentstopped** state, but is not synchronized or in the **Idling** state, but is not synchronized.

-clean
(Optional) Specifies that the VDisk that is to become a secondary is clean; any changes made on the secondary VDisk are ignored, and only changes made on the clean primary VDisk are considered when synchronizing the primary and secondary disks. The relationship must be in an Idling (connected) state for this parameter to work.

rc_rel_id | rc_rel_name
(Required) Specifies the ID or name of the relationship that you want to start in a stand-alone relationship.

Description

The **startrcrelationship** command starts a stand-alone relationship. The command fails if it is used to start a relationship that is part of a consistency group.

This command can only be issued to a relationship that is connected. For a relationship that is idling, this command assigns a copy direction (primary and secondary roles) and begins the copy process. Otherwise, this command restarts a previous copy process that was stopped either by a stop command or by some I/O error.

If the resumption of the copy process leads to a period of time when the relationship is not consistent, you must specify the **-force** parameter when you restart the relationship. This situation can arise if the relationship had been stopped, and then further input transactions had occurred on the original primary of the relationship. Even though you use of the **-force** parameter in this situation, the data on the secondary is not be useful for disaster recovery purposes because the relationship is in an inconsistent state.

In the idling state, you must provide the **-primary** parameter. In other connected states, you can provide the **-primary** parameter, but it must match the existing setting.

The **-force** parameter is required if consistency would be lost by starting a copy operation. This can occur if input transactions have occurred on either the primary or secondary VDisks since the **ConsistentStopped** or **Idling** state occurred. If the **startrcrelationship** command is issued without the **-force** parameter in such circumstances, the command fails. In general, the **-force** parameter is required if the relationship is in one of the following states:

- ConsistentStopped but not synchronized
- Idling but not synchronized

The **-force** parameter is not required if the relationship is in one of the following states:

- InconsistentStopped
- InconsistentCopying
- ConsistentSynchronized

However, the command will not fail if you do specify the **-force** parameter.

The **-clean** parameter is used when a Metro Mirror or Global Mirror relationship is started and the secondary VDisk in the relationship is assumed to be clean; any changes made on the secondary VDisk are ignored, and only changes made on the clean primary VDisk are considered when synchronizing the primary and secondary disks. The **-clean** parameter can be used in the following circumstance:

1. A relationship is created with the **-synch** parameter specified. (At this point it does not matter if the primary and secondary disks contain the same data, even though the use of the **-synch** parameter implies that this is true).
2. A **svctask stoprcrelationship** command is issued with the **-access** parameter specified. This permits access to the secondary disk. Change recording begins at the primary disk.

3. An image of the primary disk is copied and loaded on to the secondary disk. It is permissible to allow updates to the primary disk during the image copy as this image need only be a *fuzzy* image of the primary disk.
4. A **svctask startrcrelationship** command that specifies the **-primary master**, **-force**, and **-clean** parameters is issued. The auxiliary disk is marked as clean and changes on the master disk that have occurred since the relationship was stopped are copied to the auxiliary disk.
5. Once the background copy has completed, the relationship becomes consistent and synchronized.

An invocation example

```
svctask startrcrelationship rccopy1
```

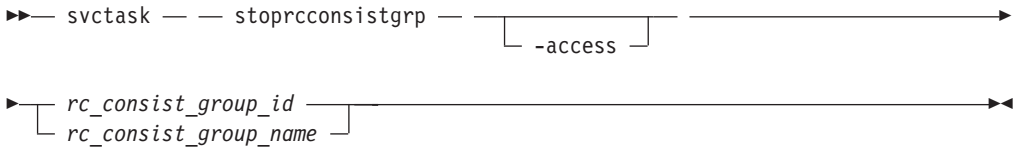
The resulting output

```
No feedback
```

stoprcconsistgrp

The **stoprcconsistgrp** command stops the copy process for a Metro Mirror or Global Mirror consistency group. This command can also be used to enable write access to the secondary VDisks in the group if the group is in a consistent state.

Syntax



Parameters

-access
(Optional) Allows write access to consistent secondary VDisks in the consistency group.

rc_consist_group_id | rc_consist_group_name
(Required) Specifies the ID or the name of the consistency group to stop all processing for.

Description

This command applies to a consistency group. You can issue this command to stop processing on a consistency group that is copying from primary VDisks to secondary VDisks.

If the consistency group is in an inconsistent state, all copy operations stop and do not resume until you issue the **svctask startrcconsistgrp** command. For a consistency group in the **ConsistentSynchronized** state, this command causes a consistency freeze.

When a consistency group is in a consistent state (for example, in the **ConsistentStopped**, **ConsistentSynchronized**, or **ConsistentDisconnected** state) you can issue the **-access** parameter with the **stoprcconsistgrp** command to enable write access to the secondary virtual disks within that group. Table 9 on page 166

shows consistency group initial and final states:

Table 9. **stoprcconsistgrp** *consistency group states.* **stoprcconsistgrp** consistency group initial and final states

Initial state	Final state	Notes
InconsistentStopped	InconsistentStopped	None.
InconsistentCopying	InconsistentStopped	None.
ConsistentStopped	ConsistentStopped	**-access** permitted
ConsistentSynchronized	ConsistentStopped	**-access** permitted
Idling	ConsistentStopped	**-access** permitted
IdlingDisconnected	unchanged	A relationship can move to the **Stopped** state when the cluster is reconnected.
InconsistentDisconnected	InconsistentStopped	These states apply to the cluster issuing the **svctask stoprcconsistgrp** command.
InconsistentDisconnected	unchanged	These states apply to the disconnected cluster.
ConsistentDisconnected	ConsistentStopped	These states apply to the cluster issuing the **svctask stoprcconsistgrp** command; the **-access** parameter is permitted.
ConsistentDisconnected	unchanged	On the disconnected cluster, the **-access** parameter is permitted.

An invocation example

```
svctask stoprcconsistgrp rccopy1
```

The resulting output

```
No feedback
```

stoprcrelationship

The **stoprcrelationship** command stops the copy process for a Metro Mirror or Global Mirror stand-alone relationship. You can also use this command to enable write access to a consistent secondary VDisk.

Syntax

Parameters

-access
(Optional) Specifies that the system allow write access to a consistent secondary VDisk.

rc_rel_id | rc_rel_name
(Required) Specifies the ID or the name of the relationship to stop all processing for.

Description

The **stoprcrelationship** command applies to a stand-alone relationship. The command is rejected if it is addressed to a relationship that is part of a consistency group. You can issue this command to stop a relationship that is copying from primary to secondary VDisks.

If the relationship is in an inconsistent state, any copy operation stops and does not resume until you issue a **svctask startrcrelationship** command. For a relationship in the **ConsistentSynchronized** state, this command causes a consistency freeze.

When a relationship is in a consistent state – in the **ConsistentStopped**, **ConsistentSynchronized**, or **ConsistentDisconnected** state – you can use the **-access** parameter to enable write access to the secondary virtual disk. Table 10 provides consistency group initial and final states.

Table 10. **stoprcrelationship** *consistency group states.* Describes **stoprcrelationship** consistency group initial and final states

Initial state	Final state	Notes
InconsistentStopped	InconsistentStopped	None.
InconsistentCopying	InconsistentStopped	None.
ConsistentStopped	ConsistentStopped	The **-access** parameter is permitted.
ConsistentSynchronized	ConsistentStopped	The **-access** parameter is permitted.
Idling	ConsistentStopped	The **-access** parameter is permitted.
IdlingDisconnected	unchanged	A relationship can move to the **stopped** state when the cluster is reconnected.
InconsistentDisconnected	InconsistentStopped	These states apply to the cluster issuing the **svctask stoprcrelationship** command.
InconsistentDisconnected	unchanged	These states apply to the disconnected cluster.
ConsistentDisconnected	ConsistentStopped	These states apply to the cluster issuing the **svctask stoprcrelationship** command. The **-access** parameter is permitted.
ConsistentDisconnected	unchanged	These states apply to the disconnected cluster. The **-access** parameter is permitted.

An invocation example

```
svctask stoprcrelationship rccopy1
```

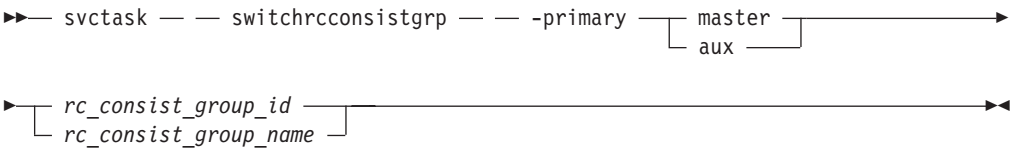
The resulting output

```
No feedback
```

switchrcconsistgrp

The **switchrcconsistgrp** command reverses the roles of the primary and secondary virtual disks (VDisks) in a Metro Mirror or Global Mirror consistency group when that consistency group is in a consistent state. All the relationships in the consistency group are affected by this change.

Syntax



Parameters

-primary master | aux
(Required) Specifies whether the master or auxiliary side of the relationships in the group will become the primary VDisks.

rc_consist_group_id | rc_consist_group_name
(Required) Specifies the ID or name of the consistency group to switch.

Description

This command applies to a consistency group. It is normally issued to reverse the roles of the primary and secondary virtual disks in a consistency group, perhaps as part of a failover process that is associated with a disaster recovery event. Write access to the former primary VDisks is lost and write access to the new primary VDisks is acquired. This command is successful when the consistency group is in a connected, consistent state, and when reversing the direction of the relationships would not lead to a loss of consistency, for example, when the consistency group is consistent and synchronized. The consistency group must be in one of the following states in order for the **switchrcconsistgrp** command to process correctly:

- ConsistentSynchronized
- ConsistentStopped and Synchronized
- Idling and Synchronized

The consistency group moves to the ConsistentSynchronized state after the successful completion of this command. If you specify the **-primary** parameter and it is the same as the current primary, the command has no effect.

An invocation example

```
svctask switchrcconsistgrp -primary aux rccopy2
```

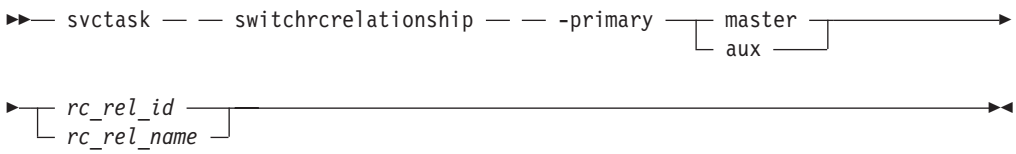
The resulting output

```
No feedback
```

switchrcrelationship

The **switchrcrelationship** command reverses the roles of primary and secondary virtual disks in a stand-alone Metro Mirror or Global Mirror relationship when that relationship is in a consistent state.

Syntax



Parameters

-primary master | aux
(Required) Specifies whether the master disk or the auxiliary disk is to be the primary.

rc_rel_id | rc_rel_name
(Required) Specifies the ID or the name of the relationship to switch.

Description

The **switchrcrelationship** command applies to a stand-alone relationship. It is rejected if it is used to try to switch a relationship that is part of a consistency group. It is normally issued to reverse the roles of the primary and secondary virtual disk in a relationship perhaps as part of a failover process during a disaster recovery event. Write access to the old primary disk is lost and write access to the new primary disk is acquired. This command is successful when the relationship is in a connected, consistent state, and when reversing the direction of the relationship does not lead to a loss of consistency; that is, when the relationship is consistent and synchronized. The relationship must be in one of the following states in order for the **switchrcrelationship** command to process correctly:

- ConsistentSynchronized
- ConsistentStopped and Synchronized
- Idling and Synchronized

The relationship moves to the **ConsistentSynchronized** state after the successful completion of this command. If you specify the **-primary** parameter with the current primary, the command has no effect.

An invocation example

```
svctask switchrcrelationship -primary master rccopy2
```

The resulting output

```
No feedback
```

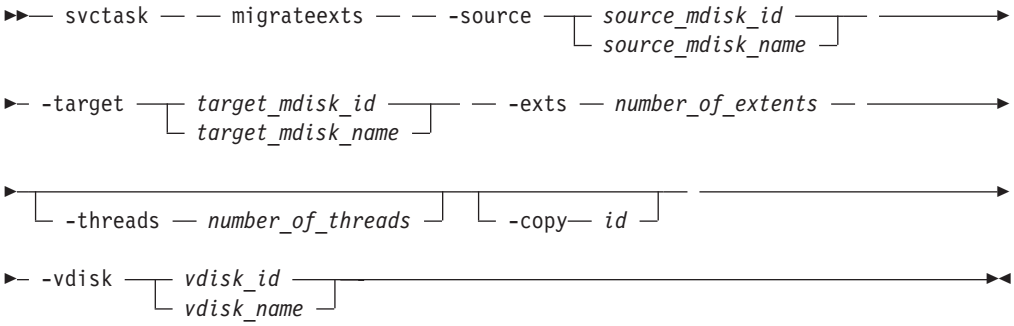
Chapter 16. Migration commands

The following commands enable you to work with migration options with the SAN Volume Controller.

migrateexts

The **migrateexts** command migrates extents from one managed disk to another.

Syntax



Parameters

-source *source_mdisk_id | source_mdisk_name*
(Required) Specifies the MDisk on which the extents currently reside.

-target *target_mdisk_id | target_mdisk_name*
(Required) Specifies the MDisk to migrate the extents to.

-exts *number_of_extents*
(Required) Specifies the number of extents to migrate.

-threads *number_of_threads*
(Optional) Specifies the number of threads to use while migrating these extents. You can specify 1 - 4 threads. The default number of threads is 4.

-copy *id*
(Required if the specified VDisk has more than one copy) Specifies the VDisk copy that the extents belong to.

-vdisk *vdisk_id | vdisk_name*
(Required) Specifies the VDisk that the extents belong to.

Description

This command migrates a given number of extents from the source virtual disk and the managed disk that contains extents that are used to make up the virtual disk. The target is a managed disk within the same managed disk group.

If a large number of extents are being migrated, you can specify 1 - 4 threads. You can issue the **svcinfo lsmigrate** command to check the progress of the migration.

The **migrateexts** command fails if there are insufficient free extents on the target managed disk. To avoid this problem, do not issue new commands that use extents until the extents migration is completed.

The **migrateexts** command fails if the target or source VDisk is offline. Correct the offline condition before attempting to migrate the VDisk.

Note: Migration activity on a single managed disk is limited to a maximum of 4 concurrent operations. This limit does not take into account whether the managed disk is the source or the destination target. If more than four migrations are scheduled for a particular managed disk, further migration operations are queued pending the completion of one of the currently running migrations. If a migration operation is stopped for any reason, a queued migration task can be started. However, if a migration is suspended, the current migration continues to use resources and a pending migration is not started. For example, the following setup is a possible initial configuration:

- MDiskGrp 1 has VDisk 1 created in it
- MDiskGrp 2 has VDisk 2 created in it
- MDiskGrp 3 has only one MDisk

With the previous configuration, the following migration operations are started:

- Migration 1 migrates VDisk 1 from MDiskGrp 1 to MDiskGrp 3, running with 4 threads.
- Migration 2 migrates VDisk 2 from MDiskGrp 2 to MDiskGrp 3, running with 4 threads.

Due to the previous limitations, the two migration operations do not always run at the same speed. MDiskGrp 3 has only one MDisk and the two migration operations have a total of 8 threads that are trying to access the one MDisk. Four threads are active. The remaining threads are in standby mode waiting to access the MDisk.

An invocation example

```
svctask migrateexts -vdisk vdisk4 -source mdisk4 -exts
 64 -target mdisk6 -threads 4
```

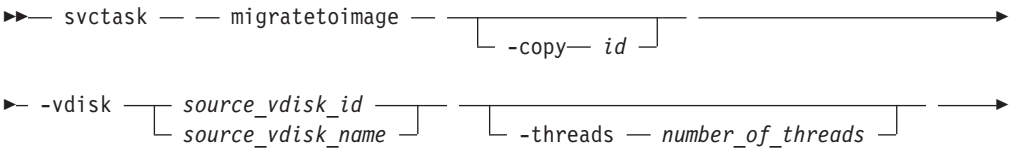
The resulting output

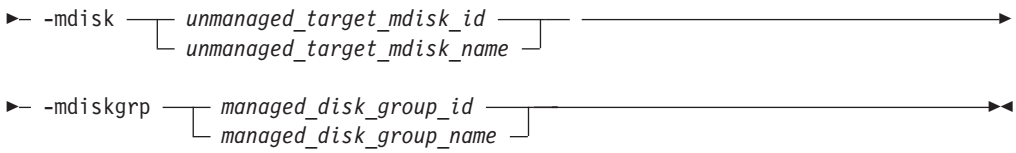
```
No feedback
```

migratetoimage

The **migratetoimage** command migrates data from a VDisk (image mode or managed mode) onto a new image mode VDisk copy. The target disk does not have to be in the same MDisk group as the source disk.

Syntax





Parameters

-vdisk *source_vdisk_id | name*
(Required) Specifies the name or ID of the source VDisk to be migrated.

-copy *id*
(Required if the specified VDisk has more than one copy) Specifies the VDisk copy to migrate from.

-threads *number_of_threads*
(Optional) Specifies the number of threads to use during the migration of extents. You can specify 1 - 4 threads. The default number of threads is 4.

-mdisk *unmanaged_target_mdisk_id | name*
(Required) Specifies the name of the MDisk to which the data must be migrated. This disk must be unmanaged and large enough to contain the data of the disk that is being migrated.

-mdiskgrp *managed_disk_group_id | name*
(Required) Specifies the MDisk group into which the MDisk must be placed, after the migration has completed.

Description

The **migratetoimage** command migrates the data of the user-specified source virtual disk onto the managed disk that is specified as the target. At completion of the command, the VDisk is classified as an image mode disk.

The managed disk that is specified as the target must be in an unmanaged state at the time that the command is run. Running this command results in the inclusion of the MDisk into the user-specified MDisk group.

The **migratetoimage** command fails if the target or source VDisk is offline. Correct the offline condition before attempting to migrate the VDisk.

The following example specifies that the user wants to migrate the data from vdisk1 onto mdisk5 and that the MDisk must be put into the MDisk group mdgrp2.

An invocation example

```
svctask migratetoimage -vdisk vdisk1 -mdisk mdisk5 -mdiskgrp mdgrp2
```

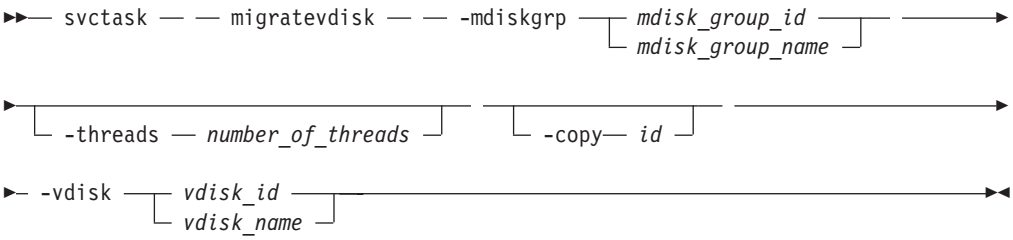
The resulting output

```
No feedback
```

migratevdisk

The **migratevdisk** command enables you to migrate an entire virtual disk from one managed disk group to another managed disk group.

Syntax



Parameters

-mdiskgrp *mdisk_group_id | mdisk_group_name*
(Required) Specifies the new managed disk group ID or name.

-threads *number_of_threads*
(Optional) Specifies the number of threads to use during the migration of these extents. You can specify 1 - 4 threads. The default number of threads is 4.

-copy *id*
(Required if the specified VDisk has more than one copy) Specifies the VDisk copy to migrate.

-vdisk *vdisk_id | vdisk_name*
(Required) Specifies the virtual disk ID or name to migrate in to a new managed disk group.

Description

The **migratevdisk** command migrates the specified virtual disk into a new managed disk group; all the extents that make up the virtual disk are migrated onto free extents in the new managed disk group.

You can issue the **svcinfo lsmigrate** command to view the progress of the migration.

The process can be prioritized by specifying the number of threads to use during the migration. Using only one thread puts the least background load on the system.

The **migratevdisk** command fails if there are insufficient free extents on the targeted managed disk group for the duration of the command. To avoid this problem, do not issue new commands that use extents until the VDisk migration is completed.

The **migratevdisk** command fails if the target VDisk or source VDisk is offline. Correct the offline condition before attempting to migrate the VDisk.

An invocation example

```
svctask migratevdisk -vdisk 4 -mdiskgrp Group0 -threads 2
```

The resulting output

```
No feedback
```

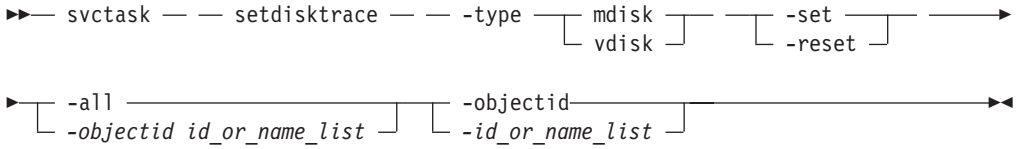
Chapter 17. Tracing commands

Tracing commands capture information that can assist you with troubleshooting managed disks and virtual disks.

setdisktrace

Use the **setdisktrace** command to set a list of disks of a given type, to include in a disk trace.

Syntax



Parameters

-type mdisk | vdisk
(Required) Specifies the object type for the disks.

-set
(Optional) Specifies the set argument. You cannot use the **-set** parameter with the **-reset** parameter.

-reset
(Optional) Specifies the reset argument. You cannot use the **-set** parameter with the **-reset** parameter.

-all
(Optional) Traces all disks of the specified type. You cannot use the **-all** parameter with the **-objectid** parameter.

-objectid *id_or_name_list*
(Optional) Specifies a list of one or more disk IDs or names. You cannot use the **-objectid** parameter with the **-all** parameter.

Description

The **setdisktrace** command marks the disks to be included in the next triggered trace.

The command is used with the **svctask settrace** command, which sets the options that result in a trace file and the data that is included in the trace file.

An invocation example

```
svctask setdisktrace -type mdisk -objectid
mdisk1:mdisk3:mdisk11:mdisk10:mdisk9:mdisk5 -reset
```

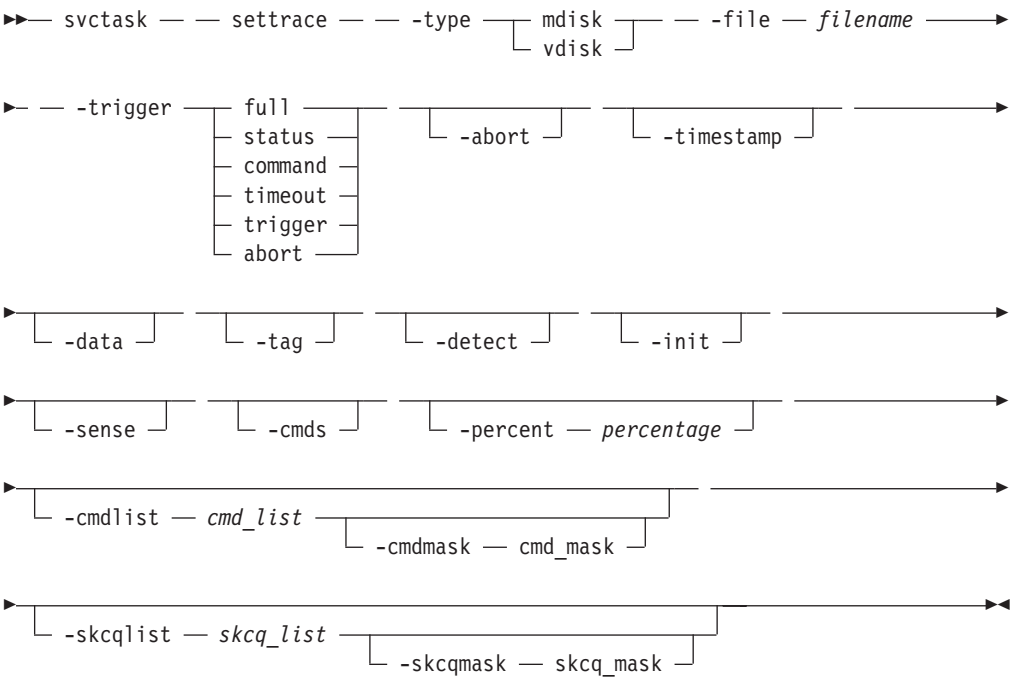
The resulting output

```
No feedback
```

settrace

The **settrace** command sets options to trace certain I/O operations through the system.

Syntax



Parameters

-type mdisk | vdisk
(Required) Specifies the type of objects to trace.

-file *filename*
(Required) Specifies the file name prefix for the trace file.

-trigger full | status | command | timeout | trigger | abort
(Required) Specifies an action for when the trace is started (triggered).

full Specifies to stop the trace when the trace buffer is full, for MDisks and VDisks.

status Sets a trigger for when the specified SCSI status (**-skcqlist**) is reported in sense data, for MDisks and VDisks.

command
Specifies a trigger for when the given SCSI command (**-cmdlist**) is sent, for MDisks and VDisks.

timeout
Sets a trigger for when a timeout occurs, for MDisks only.

trigger
Specifies to keep running until the trigger event, for MDisks only.

abort Sets a trigger for when an abnormal end occurs, for VDisks only.

-abort
(Optional) Adds abnormal ending details to the trace, for VDisks only.

-timestamp
(Optional) Adds a time-stamp to each entry in the trace. A file name is created from the prefix plus a time-stamp. The file name is in the form *prefix_AAAAAA_YYMMDD_HHMMSS*, where *AAAAAA* is the panel name of the node generating the trace file.

-data
(Optional) Adds I/O data to the trace.

-tag
(Optional) Adds CCB tags to the trace, for MDisks only.

-detect
(Optional) Adds MDisk discovery details to the trace, for MDisks only.

-init
(Optional) Adds MDisk initialization details to the trace, for MDisks only.

-sense
(Optional) Adds SCSI sense data to the trace, for VDisks only.

-cmds
(Optional) Adds commands data to the trace, for VDisks only.

-percent
(Optional) Specifies the trigger point in the trace file, which determines the amount of data to collect after the trigger point. The default value is **50**, which places the trigger point in the middle of the trace file.

-cmdlist *cmd_list*
(Optional) Adds the commands in the *cmd_list* to the trace file.

-cmdmask *cmd_mask*
(Optional) Adds the commands in the *cmd_mask* to the trace file. The **-cmdmask** parameter must be used with the **-cmdlist** parameter.

-skcqlist *skcq_list*
(Optional) Specifies an SKCQ list, which adds only those SKCQ details to the trace file.

-skcqmask *skcq_mask*
(Optional) Specifies an SKCQ mask, which adds only those SKCQ details to the trace file. The **-skcqmask** parameter must be used with the **-skcqlist** parameter.

Description

The **settrace** command sets the various I/O tracing options for managed disks or virtual disks. When the relevant disk type trace is subsequently triggered, the options specify the data to be included in the trace file.

The file name specifies a file name prefix to use when you are generating a trace file. The system appends the node panel name and a timestamp to the file name.

A maximum of 10 trace files are kept on the cluster. When the eleventh trace is made, the oldest existing trace file is overwritten.

The directory can also hold files that are retrieved from other nodes. These files are not counted. The cluster deletes the oldest file to maintain the maximum number of files.

An invocation example

```
svctask settrace -type vdisk -file tracedump -trigger abort
 -percent 100 -abort -timestamp
```

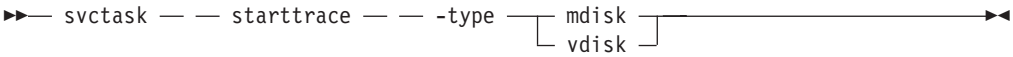
The resulting output

```
No feedback
```

starttrace

Use the **starttrace** command to begin tracing I/O operations that are based on the option currently set for the specified object type and the list of disks to trace.

Syntax



Parameters

-type mdisk | vdisk
Specifies the object type to trigger.

Description

This command starts the collection of I/O tracing information. The trace file is generated according to the options that you specified in the **svctask settrace** command. The disks that are traced are those that are identified in the list that is set by the **svctask setdisktrace** command.

The traces are written to the **/dumps/iotrace** directory. You can view the contents of this directory using the **svcinfo lsiotracedumps** command.

An invocation example

```
svctask starttrace -type vdisk
```

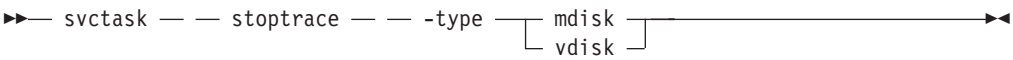
The resulting output

```
No feedback
```

stoptrace

Use the **stoptrace** command to stop tracing operations for the specified disk type.

Syntax



Parameters

-type mdisk | vdisk
(Required) Specifies the object type to stop tracing.

Description

This command stops the tracing of I/O operations for the specified object type. A a trace file is not generated if the trigger options have not been met.

An invocation example

```
svctask stoptrace -type mdisk
```

The resulting output

```
No feedback
```

Chapter 18. Attributes of the -filtervalue parameters

The **-filtervalue** parameter filters a view that is based on specific attribute values that relate to each object type. You can combine multiple filters to create specific searches, for example, **-filtervalue name=fred:status=online**. The help (**-filtervalue?**) specifies the attributes that are available for each object type.

The **-filtervalue** parameter must be specified with *attrib=value* The **-filtervalue?** and **-filtervalue** parameters cannot be specified together.

Note: The qualifier characters left bracket (<) and right bracket (>) must be enclosed within double quotation marks (""). For example, **-filtervalue vdisk_count "<"4 or port_count ">"1**. It is also valid to include the entire expression within double quotation marks. For example,**-filtervalue "vdisk_count<4"**

When an attribute requires the **-unit** parameter, it is specified after the attribute. For example, **-filtervalue capacity=24 -unit mb**. The following input options are valid for the **-unit** parameter:

- **b** (bytes)
- **mb** (Megabytes)
- **gb** (Gigabytes)
- **tb** (Terabytes)
- **pb** (Petabytes)

Table 11 provides a list of valid filter attributes, as well as descriptions, qualifiers and wildcards for each object type.

You can use the asterisk (*) character as a wildcard character when names are used. The asterisk character can be used either at the beginning or the end of a text string, but not both. Only one asterisk character can be used in a **-filtervalue** parameter.

Table 11. Valid filter attributes. Valid filter attributes for the **-filtervalue** parameter.

Object	Attribute	Valid Qualifiers	Wildcard Valid	Description
cluster	*cluster_name* or *name*	=	Yes	The cluster name.
	cluster_unique_id or *id*	=, <, <=, >, >=	No	The cluster ID.

Table 11. Valid filter attributes (continued). Valid filter attributes for the **-filtervalue** parameter.

Object	Attribute	Valid Qualifiers	Wildcard Valid	Description
node	*node_name* or *name*	=	Yes	The node name.
	id	=, <, <=, >, >=	No	The node ID.
	status	=	No	The status of the node. The following values are valid for node *status*: • **adding** • **deleting** • **online** • **offline** • **pending**
	IO_group_name	=	Yes	The I/O group name.
	IO_group_id	=, <, <=, >, >=	No	The I/O group ID.
	hardware	=	No	The following values are valid for *hardware* type: **8A4**, **8G4**, **8F2**, **8F4**, and **4F2**.
io_grp	*HWS_name* or *name*	=	Yes	The I/O group name.
	HWS_unique_id or *id*	=, <, <=, >, >=	No	The I/O group ID.
	node_count	=, <, <=, >, >=	No	The number of nodes in the I/O group.
	host_count	=, <, <=, >, >=	No	The number of hosts associated with the io_grp.
controller	*controller_id* or *id*	=, <, <=, >, >=	No	The controller ID.
mdisk	*name*	=	Yes	The name of the MDisk.
	id	=, <, <=, >, >=	No	The ID of the MDisk.
	controller_name	=	Yes	The name of the controller the MDisk belongs to.
	status	=	No	The status of the MDisk. The following values are valid for MDisk *status*: • **online** • **degraded** • **excluded** • **offline**
	mode	=	No	The mode of the MDisk. The following values are valid for MDisk *mode*: • **unmanaged** • **managed** • **image**
	mdisk_grp_name	=	Yes	The MDisk group name.
	mdisk_grp_id	=, <, <=, >, >=	No	The MDisk group ID.
	capacity	=, <, <=, >, >=	No	The capacity. Requires the **-unit** parameter.

Table 11. Valid filter attributes (continued). Valid filter attributes for the **-filtervalue** parameter.

Object	Attribute	Valid Qualifiers	Wildcard Valid	Description
mdiskgrp	*name*	=	Yes	The MDisk group name.
	storage_pool_id or *id*	=, <, <=, >, >=	No	The MDisk group ID.
	mdisk_count	=, <, <=, >, >=	No	The number of MDisks in the group.
	vdisk_count	=, <, <=, >, >=	No	The number of VDisks in the group.
	status	=	No	The status of the MDisk group. The valid input options are **online**, **degraded**, and **offline**.
	extent_size	=, <, <=, >, >=	No	The extent size. (MB)
vdisk	*vdisk_name* or *name*	=	Yes	The name of the VDisk.
	vdisk_id or *id*	=, <, <=, >, >=	No	The ID of the VDisk.
	IO_group_name	=	Yes	The name of the I/O group.
	IO_group_id	=, <, <=, >, >=	No	The ID of the I/O group.
	status	=	No	The status of the VDisk. The valid input options for VDisk status are **online**, **degraded**, and **offline**.
	mdisk_grp_name	=	Yes	The MDisk group name.
	mdisk_grp_id	=, <, <=, >, >=	No	The MDisk group ID.
	capacity	=, <, <=, >, >=	No	The capacity. Requires the -unit argument.
	type	=	No	The VDisk type. The valid value options are **seq**, **striped**, and **image**.
	FC_name	=	Yes	The FlashCopy mapping name.
	FC_id	=, <, <=, >, >=	No	The FlashCopy mapping ID.
	fc_map_count	=, <, <=, >, >=	No	The number of VDisk mappings (either source or target).
	copy_count	=, <, <=, >, >=	No	The number of VDisk mirrored copies.
	RC_name	=	Yes	The Metro Mirror relationship name.
	RC_id	=, <, <=, >, >=	No	The Metro Mirror relationship ID.
vdisk_copy	*primary*	=	No	Indicates that this copy is the primary copy. The valid values are **yes** and **no**.
	status	=	No	The status of the MDisk group. Valid values are **online**, **degraded**, or **offline**.
	sync	=	No	Indicates whether the VDisk copy is synchronized. Valid values are **true** or **false**.
	mdisk_grp_name	=	Yes	The name of the MDisk group.
	mdisk_grp_id	=, <, <=, >, >=	No	The ID of the MDisk group.
	type	=	No	The type of the VDisk copy. The valid values are **seq**, **striped**, or **image**.

Table 11. Valid filter attributes (continued). Valid filter attributes for the **-filtervalue** parameter.

Object	Attribute	Valid Qualifiers	Wildcard Valid	Description
se_vdiskcopy	*mdisk_grp_id*	=, <, <=, >, >=	No	The ID of the MDisk group.
	mdisk_grp_name	=	Yes	The name of the MDisk group.
	overallocation	=	No	The percentage of overallocation, which is displayed as a number.
	autoexpand	=	No	Autoexpand flags. The valid values are **on** and **off**.
	grainsize	=, <, <=, >, >=	No	Space-efficient grain size. The valid values are **32**, **64**, **128**, or **256**.
host	*host_name* or *name*	=	Yes	The host name.
	host_id or *id*	=, <, <=, >, >=	No	The host ID.
	port_count	=, <, <=, >, >=	No	The number of ports.
	iogrp_count	=, <, <=, >, >=	No	The number of I/O groups that are associated with the host.
fcmap	*FC_mapping_name* or *name*	=	Yes	The FlashCopy mapping name.
	FC_id or *id*	=, <, <=, >, >=	No	The FlashCopy mapping ID.
	source_vdisk_name	=	Yes	The source VDisk name.
	source_vdisk_id	=, <, <=, >, >=	No	The source VDisk ID.
	target_vdisk_name	=	Yes	The target VDisk name.
	target_vdisk_id	=, <, <=, >, >=	No	The target VDisk ID.
	group_name	=	Yes	The consistency group name.
	group_id	=, <, <=, >, >=	No	The consistency group ID.
	status	=	No	The mapping status. The following values are valid for fcmap *status*: • **idle_or_copied** • **preparing** • **prepared** • **copying** • **stopped** • **suspended** • **stopping** • **empty**
	copy_rate	=, <, <=, >, >=	No	The background copy rate.

Table 11. Valid filter attributes (continued). Valid filter attributes for the **-filtervalue** parameter.

Object	Attribute	Valid Qualifiers	Wildcard Valid	Description
fcconsistgrp	*name*	=	Yes	The consistency group name.
	FC_group_id or *id*	=, <, <=, >, >=	No	The consistency group ID.
	status	=	No	The consistency group status. The following values are valid for fcconsistgrp *status*: • **idle_or_copied** • **preparing** • **prepared** • **copying** • **stopped** • **suspended** • **stopping** • **empty**

Table 11. Valid filter attributes (continued). Valid filter attributes for the **-filtervalue** parameter.

Object	Attribute	Valid Qualifiers	Wildcard Valid	Description
rcrelationship	*RC_rel_id* or *id*	=, <, <=, >, >=	No	The Metro Mirror relationship ID.
	RC_rel_name or *name*	=	Yes	The Metro Mirror relationship name.
	master_cluster_id	=, <, <=, >, >=	No	The master cluster ID.
	master_cluster_name	=	Yes	The master cluster name.
	master_vdisk_id	=, <, <=, >, >=	No	The master VDisk ID.
	master_vdisk_name	=	Yes	The master VDisk name.
	aux_cluster_id	=, <, <=, >, >=	No	The aux cluster ID.
	aux_cluster_name	=	Yes	The aux cluster name.
	aux_vdisk_id	=, <, <=, >, >=	No	The aux VDisk ID.
	aux_vdisk_name	=	Yes	The aux VDisk name.
	primary	=	No	The relationship primary. The following values are valid for *primary*: • **master** • **aux**
	consistency_group_id	=, <, <=, >, >=	No	The Metro Mirror consistency group ID.
	consistency_group_name	=	Yes	The Metro Mirror consistency group name.
	state	=	Yes	The relationship state. The following values are valid for *state*: • **inconsistent_stopped** • **inconsistent_copying** • **consistent_stopped** • **consistent_synchronized** • **idling** • **idling_disconnected** • **inconsistent_disconnected** • **consistent_disconnected**
	progress	=, <, <=, >, >=	No	The progress of the initial background copy (synchronization) for the relationship.

Table 11. Valid filter attributes (continued). Valid filter attributes for the **-filtervalue** parameter.

Object	Attribute	Valid Qualifiers	Wildcard Valid	Description
rcconsistgrp	*group_id or id*	=, <, <=, >, >=	No	The consistency group ID.
	name	=	Yes	The consistency group name.
	master_cluster_id	=, <, <=, >, >=	No	The master cluster ID.
	master_cluster_name	=	Yes	The master cluster name.
	aux_cluster_id	=, <, <=, >, >=	No	The aux cluster ID.
	aux_cluster_name	=	Yes	The aux cluster name.
	primary	=	No	The consistency group primary. The following values are valid for *primary*: • **master** • **aux**
	state	=	No	The consistency group state. The following values are valid for *state*: • **inconsistent_stopped** • **inconsistent_copying** • **consistent_stopped** • **consistent_synchronized** • **idling** • **idling_disconnected** • **inconsistent_disconnected** • **consistent_disconnected** • **empty**
	relationship_count	=, <, <=, >, >=	No	The relationship count.

Chapter 19. Overview of the list dumps commands

The list dumps commands return a list of dumps in the appropriate directory.

SAN Volume Controller dumps are contained in the following directory structure:

- **/dumps**
- **/dumps/audit**
- **/dumps/cimom**
- **/dumps/elogs**
- **/dumps/feature**
- **/dumps/iostats**
- **/dumps/iotrace**

Software upgrade packages are contained in the **/home/admin/upgrade** directory. These directories exist on every node in the cluster.

An audit log keeps track of action commands that are issued through an SSH session or from the SAN Volume Controller Console. To list the audit log files in the **/dumps/audit** directory on the current configuration node or on the designated node, issue the **svcinfo lsauditlogdumps** command. To list a specified number of the most recently audited commands, issue the **svctask catauditlog** command. To dump the contents of the audit log to a file on the current configuration node, issue the **svctask dumpauditlog** command. This command also clears the contents of the audit log.

Dumps contained in the **/dumps/cimom** directory are created by the CIMOM (Common Information Model Object Manager) that runs on the SAN Volume Controller cluster. These files are produced during normal operations of the CIMOM. To list all the files in the **/dumps/cimom** directory, issue the **svcinfo lscimomdumps** command.

Dumps that are contained in the **/dumps/elogs** directory are dumps of the contents of the error and event log at the time that the dump was taken. An error or event log dump is created by using the **svctask dumperrlog** command. This dumps the contents of the error or event log to the **/dumps/elogs** directory. If no file name prefix is supplied, the default **errlog_** is used. The full default file name is **errlog_***NNNNNN_YYMMDD_HHMMSS*, where *NNNNNN* is the node front panel name. If the command is used with the **-prefix** parameter, the prefix value is used instead of **errlog**. To list all dumps in the **/dumps/elogs** directory, issue the **svcinfo lserrlogdumps** command.

Dumps contained in the **/dumps/feature** directory are dumps of the featurization log. A featurization log dump is created by using the **svctask dumpinternallog** command. This dumps the contents of the featurization log to the **/dumps/feature** directory to a file called **feature.txt**. Only one of these files exists, so every time the **svctask dumpinternallog** command is run, this file is overwritten. To list all dumps in the **/dumps/feature** directory, issue the **svcinfo lsfeaturedumps** command.

Dumps that are contained in the **/dumps/iostats** directory are dumps of the I/O statistics for disks on the cluster. An I/O statistics dump is created by using the **svctask startstats** command. As part of this command, you can specify a time

interval for the statistics to be written to the file; the default is 15 minutes. Every time the time interval is encountered, the I/O statistics that have been collected are written to a file in the **/dumps/iostats** directory. The file names that are used for storing I/O statistics dumps are **m_stats**_*NNNNNN_YYMMDD_HHMMSS*, **m_stats**_*NNNNNN_YYMMDD_HHMMSS*, and **v_stats**_*NNNNNN_YYMMDD_HHMMSS*, where *NNNNNN* is the node front panel name, depending on whether the statistics are for MDisks or VDisks. To list all dumps in the **/dumps/iostats** directory, issue the **svcinfo lsiostatsdumps** command.

Dumps that are contained in the **/dumps/iotrace** directory are dumps of I/O trace data. The type of data that is traced depends on the options specified by the **svctask settrace** command. The collection of the I/O trace data is started by using the **svctask starttrace** command. The I/O trace data collection is stopped when the **svctask stoptrace** command is used. It is when the trace is stopped that the data is written to the file. The file name is *prefix_NNNNNN_YYMMDD_HHMMSS*, where *prefix* is the value entered for the **-filename** parameter in the **svctask settrace** command, and *NNNNNN* is the node front panel name. To list all dumps in the **/dumps/iotrace** directory, issue the **svcinfo lsiotracedumps** command.

Dumps that are contained in the **/dumps** directory result from application abends. Such dumps are written to the **/dumps** directory. The default file names are **dump**.*NNNNNN.YYMMDD.HHMMSS*, where *NNNNNN* is the node front panel name. In addition to the dump file, there might be some trace files written to this directory, are named *NNNNNN*.**trc**.

To list all dumps in the **/dumps** directory, issue the **svcinfo ls2145dumps** command.

The final option available in the list dumps command series is the **svcinfo lssoftwaredumps** command. This command lists the contents of the **/home/admin/upgrade** directory; files are copied to this directory during software upgrades.

All of the list dumps commands can accept a node identifier as input. If this identifier is not specified, the list of files on the current configuration node are displayed. If the node identifier is specified, the list of files on that node are displayed.

Because files can only be copied from the current configuration node (using secure copy), you can issue the **svctask cpdumps** command to copy the files from a nonconfiguration node to the current configuration node.

Chapter 20. Information commands

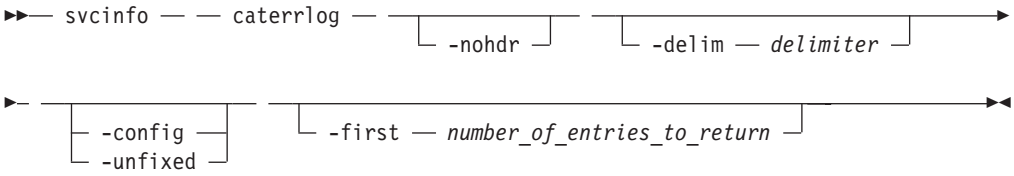
The following commands enable you display specific types of SAN Volume Controller information.

Note: IDs are assigned at run-time by the system and cannot be relied upon to be the same after configuration restoration. Therefore, use object names instead of IDs whenever possible.

caterrlog

The **caterrlog** command displays the contents of the cluster error log and event log.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-config
(Optional) Specifies to list the configuration events.

-unfixed
(Optional) Specifies to list the unfixed errors.

-first *number_of_entries_to_return*
(Optional) Displays the first *x* number of entries in the log, where *x* is the number that is entered by the user. The value of *x* can be 1 - 256.

Description

This command displays a list of the specified error log entries. When no parameters are used, all error log entries are listed, to a maximum of 256 entries.

The list can be filtered to only include configuration events or unfixed errors by specifying the **-config** or **-unfixed** parameters.

The **-first** parameter results in the display of the first *x* number of records, where *x* is the number that is entered for the **-first** parameter.

An invocation example

```
svcinfo caterrlog -delim :
```

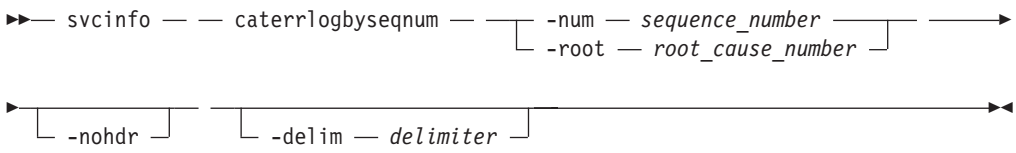
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error_code
:copy_id
0:cluster:no:no:6:node1:100:100:030407052547:030407052547:1:00981001
0:fc_card:no:no:1:node1:101:101:030407052547:030407052547:1:00073001
1:node:no:no:1:node1:102:102:030407052547:030407052547:1:00074001
0:cluster:no:no:6:node1:103:100:030407052547:030407052547:1:00981001
1:fc_card:no:no:1:node1:104:104:030407052632:030407052632:1:00073003
0:node:no:no:6:node1:105:105:030407082202:030407082717:2:00980500
2:remote:no:no:6:n/a:106:106:030407090117:030407090117:1:00985002
1:node:no:no:5:node1:0:0:030407052546:030407052546:1:00990383
0:cluster:no:no:5:node1:0:0:030407080630:030407080630:1:00990117
0:mdisk_grp:no:no:5:node1:0:0:030407081610:030407081610:1:00990148
128:mdisk_grp:no:no:5:node1:0:0:030407081610:030407081610:1:00990173
1:mdisk_grp:no:no:5:node1:0:0:030407081619:030407081619:1:00990148
0:vdisk:no:no:5:node1:0:0:030407081836:030407081836:1:00990169:0
1:vdisk:no:no:5:node1:0:0:030407081843:030407081843:1:00990169:0
0:vdisk:no:no:5:node1:0:0:030407081854:030407081854:1:00990169:0
0:vdisk:no:no:5:node1:0:0:030407082015:030407082015:1:00990169:0
0:vdisk:no:no:5:node1:0:0:030407082145:030407082145:1:00990169:0
0:vdisk:no:no:5:node1:0:0:030407082148:030407082148:1:00990169:0
0:vdisk:no:no:5:node1:0:0:030407082158:030407082158:1:00990169:0
1:vdisk:no:no:5:node1:0:0:030407082213:030407082213:1:00990169:0
0:host:no:no:5:node1:0:0:030407082441:030407082441:1:00990106
1:host:no:no:5:node1:0:0:030407082457:030407082457:1:00990106
2:host:no:no:5:node1:0:0:030407082523:030407082523:1:00990106
0:flash:no:no:5:node1:0:0:030407082704:030407082704:1:00990184
1:node:no:no:5:node1:0:0:030407082716:030407082716:1:00990501
1:node:no:no:5:node1:0:0:030407082722:030407082722:1:00990501
1:fc_const_grp:no:no:5:node1:0:0:030407083141:030407083141:1:00990204
2:fc_const_grp:no:no:5:node1:0:0:030407083143:030407083143:1:00990204
3:fc_const_grp:no:no:5:node1:0:0:030407083145:030407083145:1:00990204
0:flash:no:no:5:node1:0:0:030407083318:030407083318:1:00990185
0:flash:no:no:5:node1:0:0:030407083355:030407083355:1:00990185
0:flash:no:no:5:node1:0:0:030407085753:030407085753:1:00990185
1:remote:no:no:5:node1:0:0:030407085932:030407085932:1:00990225
2:vdisk:no:no:5:node1:0:0:030407085959:030407085959:1:00990169:0
3:vdisk:no:no:5:node1:0:0:030407090004:030407090004:1:00990169:0
4:vdisk:no:no:5:node1:0:0:030407090013:030407090013:1:00990169:0
2:remote:no:no:5:node1:0:0:030407090106:030407090106:1:00990225
255:rc_const_grp:no:no:5:node1:0:0:030407090323:030407090323:1:00990240
254:rc_const_grp:no:no:5:node1:0:0:030407090327:030407090327:1:00990240
253:rc_const_grp:no:no:5:node1:0:0:030407090333:030407090333:1:00990240
2:remote:no:no:5:node1:0:0:030407090442:030407090442:1:00990226
1:vdisk:no:no:5:node1:0:0:030407090820:030407090820:1:00990182:0
3:vdisk:no:no:5:node1:0:0:030407090825:030407090825:1:00990182:0
```

caterrlogbyseqnum

The **caterrlogbyseqnum** command displays all the errors with the sequence number or root cause number that has been specified by the user.

Syntax



Parameters

-num *sequence_number*
(Required if **-root** is not specified) Specifies the sequence number to view.

-root *root_cause_number*
(Required if **-num** is not specified) Specifies the root sequence number. All errors that are marked with this root cause are displayed.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command displays a single error log entry, determined by the sequence number that is specified as the **-num** parameter.

If the **-root** parameter is used, the log is searched for all entries that are marked with a root cause sequence number, as specified. A list of all entries marked with this root cause is displayed.

An invocation example

```
svcinfo caterrlogbyseqnum -num 100 -delim :
```

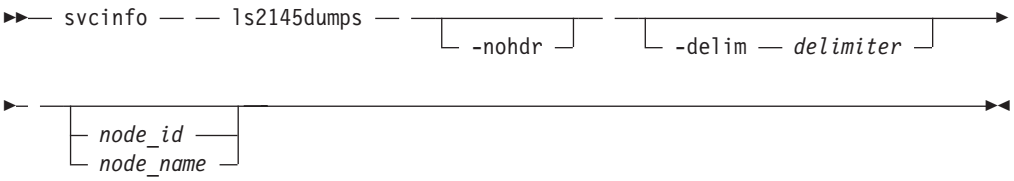
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:
error_code:copy_id
0:cluster:no:no:6:node1:100:100:030407052547:030407052547:1:00981001:
```

ls2145dumps

The **ls2145dumps** command obtains a list of node assert dumps and the associated output files from the **/dumps** directory.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

node_id | *node_name*
(Optional) Displays a list of the available dumps of the given type for the specified node ID or name. If you do not specify a node, the available dumps on the configuration node are listed.

Description

This command displays a list of node assert dumps and associated output files that are available. These dumps are created as a result of the assertion of a node. If you do not specify a node, the dumps that are available on the configuration node are listed. The command displays files from the **/dumps** directory.

An invocation example

```
svcinfo ls2145dumps -delim :
```

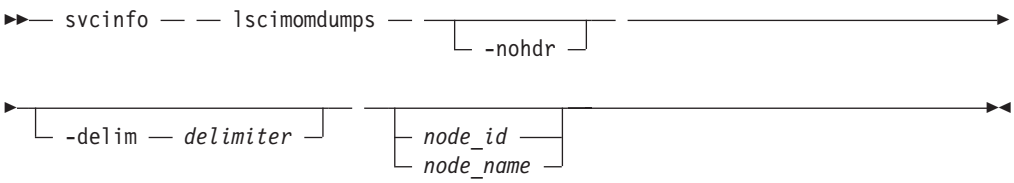
The resulting output

```
id:2145_filename
0:000108.trc.old
1:dump.000108.030328.144007
2.000108.trc
```

lscimomdumps

Use the **lscimomdumps** command to display a list of files in the **/dumps/cimom** directory.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, then the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

node_id | *node_name*
(Optional) Specifies the node ID or name to list the available dumps of the given type. If you do not specify a node, the dumps that are available on the configuration node are listed.

Description

This command displays a list of Common Information Model object manager (CIMOM) dumps. These dumps are created when you use a CIM client with the CIMOM of the SAN Volume Controller cluster. If you do not specify a node, the dumps that are available on the configuration node are listed. The command displays files from the **/dumps/cimom** directory.

An invocation example

```
svcinfo lscimomdumps
```

The resulting output

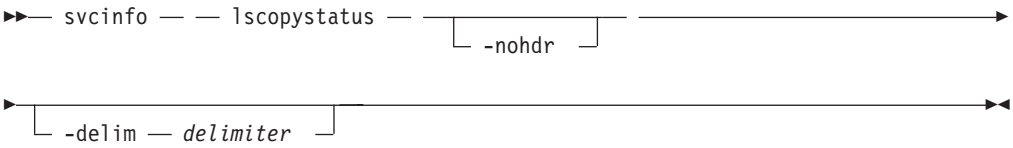
```
id               cimom_filename
0                mkrepositorylog.004565
1                PegasusTrace.004565
```

```
2               PegasusStandard.004565
3               PegasusAudit.004565
4               PegasusError.004565
5               PegasusDebug.004565
```

lscopystatus

Use the **lscopystatus** command to determine whether any file copies are currently in progress.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command displays an indicator that shows if a file copy is currently in progress. Only one file can be copied in the cluster at a time.

An invocation example

```
svcinfo lscopystatus
```

The resulting output

```
status
 active
```

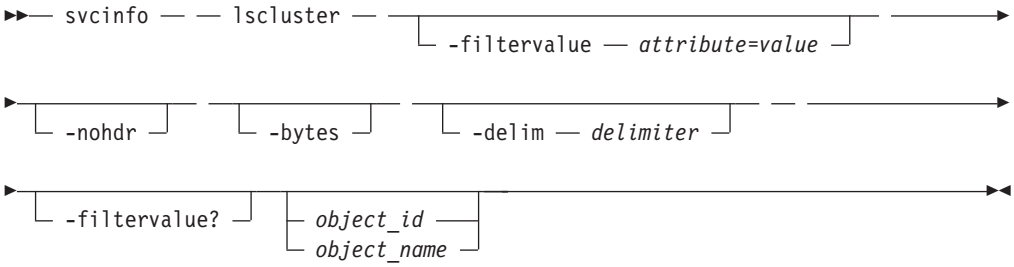
lscluster

The **lscluster** command provides a report that you can designate as a concise list or a detailed list of clusters. The list provides possible values that are applicable to the attributes that are displayed as data in the output views.

The list report can provide two styles of report:

- A list containing concise information about all clusters. (Each entry in the list corresponds to a single cluster.)
- The detailed information about a single, user-specified cluster.

Syntax



Parameters

-filtervalue *attribute=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are displayed. If a capacity is specified, the units must also be included.

Note: Some filters allow the asterisk character (*) when you enter the command. The following rules apply to the use of wildcard characters with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard.
- When you use a wildcard, you must enclose the filter entry within double quotation marks (""), as follows:

```
svcinfo lscluster -filtervalue  "name=md*"
```

-filtervalue?
(Optional) displays a list of filters that can be applied against this view. The following filter attributes are valid for the **lscluster** command:

- **cluster_name**
- **cluster_unique_id**
- **id**
- **name**

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-bytes
(Optional) Specifies that you want the report to display all capacities as bytes.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | object_name
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is displayed and any value that you specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id | object_name* parameter, the concise view of all objects that match the filtering requirements that are specified by the **-filtervalue** parameter are displayed.

Description

This command displays a concise list or a detailed view of a cluster.

Table 12 provides the attribute values that can be displayed as output view data.

Table 12. Attribute values. Describes lscluster command output view attribute values

Attribute	Possible Values
location	local, remote
statistics status	on, off
SNMP setting	none, all, hardware_only
email_state	running, stopped, invalid

Information about the remote cluster is reported by the **lscluster** command if the **mkpartnership** command has been issued from the local cluster to the remote cluster; for example, if the partnership has been at least partially established from the local cluster.

You can issue the **svcinfo lscluster** command to display a concise view of the cluster.

```
svcinfo lscluster -delim : 10030a007e5
```

where **10030a007e5** is the cluster name.

Concise view shows the fields described for remote clusters only; if the cluster **Location** is **local**, then **Partnership** and **Bandwidth** do not apply (and are not defined or provided). For a remote cluster, these fields indicate the following information:

- **Location**: remote
- **Partnership**:

partially_configured
The **mkpartnership** command has only been issued from the local cluster to the remote cluster.

fully_configured
The **mkpartnership** command has been issued in both directions.

- **Bandwidth**: MBps (the available bandwidth on the intercluster a link for background copy)

A concise invocation example

```
svcinfo lscluster -delim ;
```

The concise resulting output

```
id;name;location;partnership;bandwidth;cluster_IP_address;cluster_service_ip_address;
cluster_IP_address_6;cluster_service_ip_address_6;id_alias
1521071282978998;cluster1;local;;;9.20.165.16;9.20.165.17;0000:0000:0000:0000:0000:
0000:0000:0000;0000:0000:0000:0000:0000:0000:0000:0000;1521071282978998
1578946783290101;cluster2;remote;partial;50;9.20.165.20;9.20.165.21;0000:0000:0000:
0000:0000:0000:0000:0000;0000:0000:0000:0000:0000:0000:0000:0000;1578946783290101
```

A detailed invocation example

```
svcinfo lscluster -delim : cluster1
```

The detailed resulting output

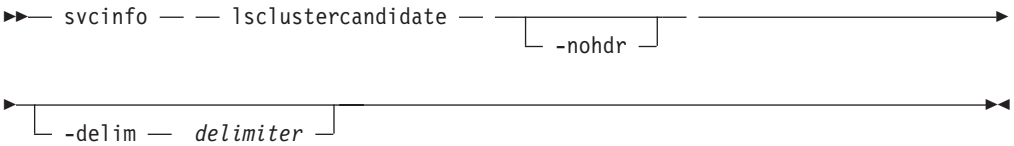
```
id:1521071282978998
name:cluster1
location:local
partnership:fully_configured_stopped
bandwidth:
cluster_IP_address:9.20.165.16
cluster_service_IP_address:9.20.165.17
total_mdisk_capacity:59.8GB
space_in_mdisk_grps:0
space_allocated to vdisks:0
total_free_space:59.8GB
statistics_status:on
statistics_frequency:300
required_memory:1280
cluster_locale:en_US
SNMP_setting:snmp all
SNMP_community:
SNMP_server_IP_address:9.20.165.18
subnet_mask:255.255.255.0
default_gateway:9.20.165.1
time_zone:522 UTC
email_setting:none
email_id:
code_level:1.20abcG
FC_Port_Speed:1Gb
console_IP:9.20.165.12:123
id_alias:1521071282978998
gm_link_tolerance:60
gm_inter_cluster_delay_simulation:0
gm_intra_cluster_delay_simulation:0
email_server:8.53.26.131
email_server_port:25
email_reply:fred@mycompany.com
email_contact:Fred Higgins
email_contact_primary:01202 123456
email_contact_alternate:44-202-876543-4455
email_contact_location:London Thames Bank
email_state:running
email_user_count:2
```

```
inventory_mail_interval:0
cluster_IP_address_6:
cluster_service_IP_address_6:
prefix_6:
default_gateway_6:
console_IP_6:
email_server_6:
total_vdiskcopy_capacity:40.00GB
total_used_capacity:22.50GB
total_overallocation:67
total_vdisk_capacity:30.00GB
cluster_ntp_ip_address:
```

lsclustercandidate

The **lsclustercandidate** command lists the clusters that are available for setting up a two-cluster partnership. This is a prerequisite for creating intercluster Metro or Global Mirror relationships.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command displays a list of clusters that are available as candidate partner clusters to form a Metro Mirror or Global Mirror Partnership between two clusters.

An invocation example

```
svcinfo lsclustercandidate
```

The resulting output

```
id               configured     cluster_name
0000010034E0F430 no             ldcluster26
```

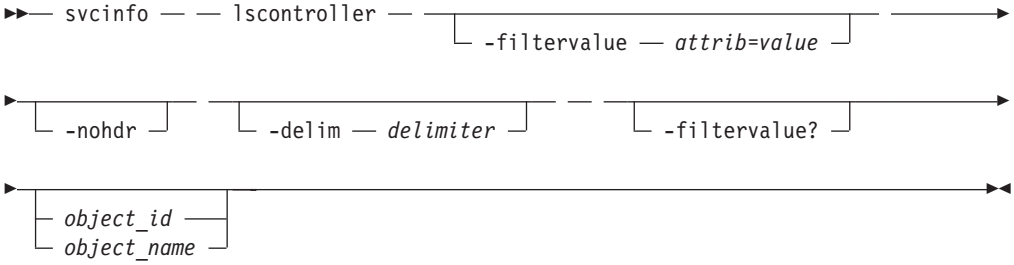
lscontroller

The **lscontroller** command returns a concise list, or a detailed view, of controllers that are visible to the cluster.

The list report style can be used to obtain two styles of report:

- A list containing concise information about controllers. (Each entry in the list corresponds to a single controller.)
- The detailed information about a single, user-specified controller.

Syntax



Parameters

-filtervalue *attrib=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are returned. If a capacity is specified, the units must also be included.

Note: Some filters allow the use of a wildcard when you enter the command. The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard, which must be the first or last character in the string.
- When using a wildcard, enclose the filter entry within double quotation marks (""), as follows:

```
svcinfo lscontroller -filtervalue "name=md*"
```

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-filtervalue?
(Optional) Displays the valid filter attributes. The following filter attributes for the **svcinfo lscontroller** command are valid:

- controller_id
- id

object_id | object_name
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id | object_name* parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

Description

This command returns a concise list, or a detailed view, of controllers visible to the cluster.

The following values are applicable to the data in the output views:
`degraded` **`no, yes`**

To differentiate the name of a storage controller from the name shown on the cluster, list the storage controllers by issuing the **svcinfo lscontroller** command. Record the controller name or ID for the controller that you want to determine. For the controller in question, issue the **svcinfo lscontroller** *controller name | id* command, where *controller name | id* is the controller name or ID. Record the worldwide node name (WWNN) for the controller. You can use the WWNN to determine the actual storage controller by launching the native controller user interface, or by using the command line tools it provides to verify the actual controller that has the WWNN.

Each MDisk corresponds to a single RAID array or to a single partition on a given RAID array. Each RAID controller defines a LUN number for the disk. The LUN number and controller name or ID are required to determine the relationship between MDisks and RAID arrays or partitions. Take the following steps to determine the relationship between MDisks and RAID arrays or LUNs:

1. Show the detailed view of the given MDisk *mdiskname* by issuing the **svcinfo lsmdisk** *mdiskname* command, where *mdiskname* is the name of the MDisk. Record the controller name or controller ID and controller LUN number.
2. Show the detailed view of the controller by issuing the **svcinfo lscontroller** *controller name | id* command, where *controller name | id* is the name or ID of the controller. Record the vendor ID, product ID, and WWNN, and use them to determine the object that is being presented to the MDisk.
3. From the native user interface for the given controller, list the LUNs and match the LUN number to determine the exact RAID array or partition that corresponds to the MDisk.

   Notes:

   a. The *path_count* value is the number of MDisks using the controller port multiplied by the number of nodes in the cluster.
   b. The *max_path_count* value is the highest possible value that the *path_count* can reach for the controller type.

c. The *allowquorum* parameter value shows whether the controller is currently enabled to support quorum disks.

A concise invocation example

```
svcinfo lscontroller -delim :
```

The concise resulting output

```
id:controller_name:ctrl_s/n:vendor_id:product_id_low:product_id_high
7:controller7:3EK0J5Y8:SEAGATE :ST373405:FC
8:controller8:3EK0J6CR:SEAGATE :ST373405:FC
9:controller9:3EK0J4YN:SEAGATE :ST373405:FC
10:controller10:3EK0GKGH:SEAGATE :ST373405:FC
11:controller11:3EK0J85C:SEAGATE :ST373405:FC
12:controller12:3EK0JBR2:SEAGATE :ST373405:FC
13:controller13:3EKYNJF8:SEAGATE :ST373405:FC
14:controller14:3EK0HVTM:SEAGATE :ST373405:FC
```

A detailed invocation example

```
svcinfo lscontroller -delim = 7
```

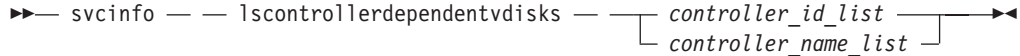
The detailed resulting output

```
id=7
controller_name=controller7
WWNN=20000004CF2412AC
mdisk_link_count=1
max_mdisk_link_count=1
degraded=no
vendor_id=SEAGATE
product_id_low=ST373405
product_id_high=FC
product_revision=0003
ctrl_s/n=3EK0J5Y8
allow_quorum=no
WWPN=22000004CF2412AC
path_count=1
max_path_count=1
WWPN=21000004CF2412AC
path_count=0
max_path_count=0
```

lscontrollerdependentvdisks

The **lscontrollerdependentvdisks** command lists the VDisks that are dependent on the specified controller.

Syntax



Parameters

controller_id_list | *controller_name_list*
Specifies one or more controller IDs, controller names, or both. Separate multiple controllers using the colon character (**:**).

Description

The **lscontrollerdependentvdisks** command lists the VDisks that are dependent on the status of the specified controllers. If a controller goes offline, the dependent VDisks also go offline. Before taking a controller offline for maintenance, you can use the command to ensure that you do not lose access to any VDisks.

If you have multiple controllers configured as a single subsystem, you must specify all of the controllers in the subsystem, using a single command invocation.

The **lscontrollerdependentvdisks** command also checks for quorum disks on the specified controller list. If any quorum disks are on the specified controller list, the command returns an error. All quorum disks must be moved before performing any maintenance. After moving quorum disks, reissue the command to list the dependent VDisks.

Note: The command lists the VDisks that are dependent on the controllers at the time the command is run; subsequent changes to your system require rerunning the command.

An invocation example

```
svcinfo lscontrollerdependentvdisks controller0
```

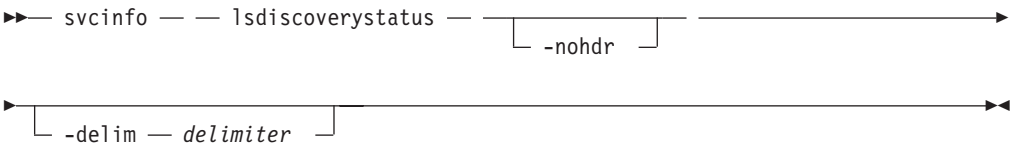
The resulting output

```
vdisk_id vdisk_name
0 vdisk0
1 vdisk1
2 vdisk2
```

lsdiscoverystatus

Use the **lsdiscoverystatus** command to determine whether a discovery operation is in progress.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by

a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command displays one of the following results:

active There is a discovery operation in progress at the time that the command is issued.

inactive
There are no discovery operations in progress at the time that the command is issued.

An invocation example

```
svcinfo lsdiscoverystatus
```

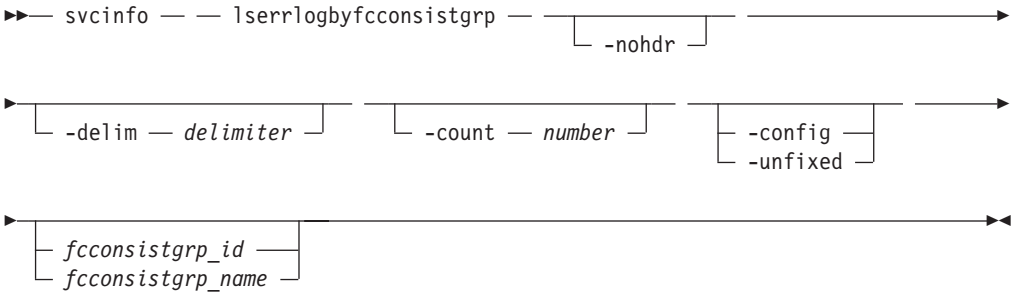
The resulting output

```
status
inactive
```

lserrlogbyfcconsistgrp

The **lserrlogbyfcconsistgrp** command displays errors and events in the log that are related to FlashCopy consistency groups.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the

-delim parameter is a one-byte character. If you enter **-delim :** on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

fcconsistgrp_id | *fcconsistgrp_name*
(Optional) Specifies the object ID that filters the log.

Description

This command displays a list of the errors and events in the log that are related to FlashCopy consistency groups. The list can be filtered further by specifying an object ID or object name. The list displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly, the last *x* number of entries for a given object type or object ID can be listed.

Note: Although there is an object type of *unknown* is displayed in the error log, there is no command available to filter this object type.

An invocation example

```
svcinfo lserrlogbyfcconsistgrp -delim :
```

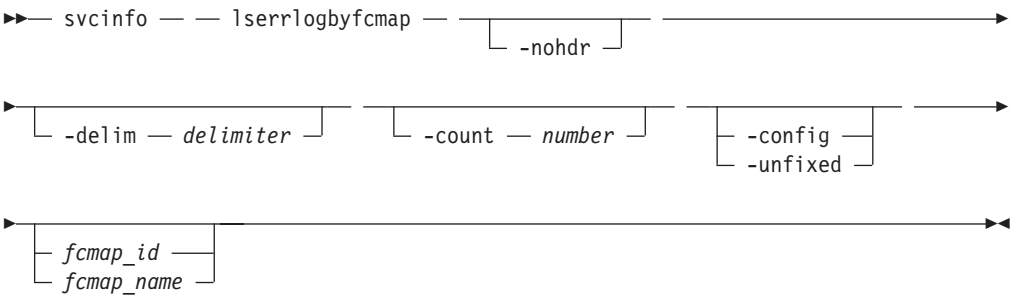
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error_code
3:fc_const_grp:no:no:5:node1:0:0:030407083145:030407083145:1:00990204
2:fc_const_grp:no:no:5:node1:0:0:030407083143:030407083143:1:00990204
1:fc_const_grp:no:no:5:node1:0:0:030407083141:030407083141:1:00990204
```

lserrlogbyfcmap

The **lserrlogbyfcmap** command displays a list of the errors and events in the log that are related to FlashCopy mappings.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed even if the **-nohdr** parameter is specified.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

fcmap_id | fcmap_name
(Optional) Specifies the object ID that filters the log.

Description

This command displays a list of the errors and events in the log that are related to FlashCopy mappings. The list can be filtered further by specifying a specific object ID or name. This list displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly, the last *x* entries against a given object type or object ID can be listed.

Note: Although an object type of *unknown* is displayed in the error log, there is no available command to filter this object type.

An invocation example

```
svcinfo lserrlogbyfcmap -delim :
```

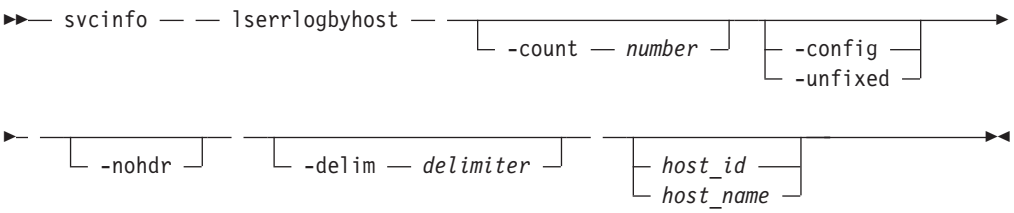
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error_code
0:flash:no:no:5:node1:0:0:030407085753:030407085753:1:00990185
0:flash:no:no:5:node1:0:0:030407083355:030407083355:1:00990185
0:flash:no:no:5:node1:0:0:030407083318:030407083318:1:00990185
0:flash:no:no:5:node1:0:0:030407082704:030407082704:1:00990184
```

lserrlogbyhost

The **lserrlogbyhost** command displays a list of the errors and events in the log that are related to hosts.

Syntax



Parameters

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed even if the **-nohdr** parameter is specified.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

host_id | *host_name*
(Optional) Specifies the object ID that filters the log.

Description

This command displays a list of the errors and events in the log that are related to hosts. The list can be filtered further by specifying a specific object ID or name. This list displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly, the last *x* entries against a given object type or object ID can be listed.

Note: Although an object type *unknown* is displayed in the error log, there is no available command to filter this object type.

An invocation example

```
svcinfo lserrlogbyhost -delim :
```

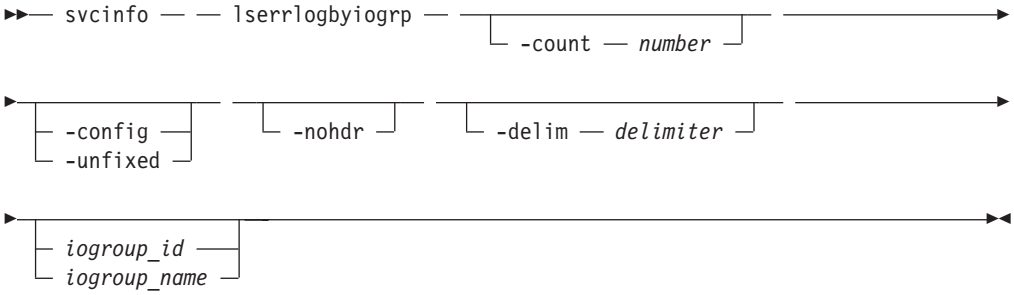
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error_code
2:host:no:no:5:node1:0:0:030407082523:030407082523:1:00990106
1:host:no:no:5:node1:0:0:030407082457:030407082457:1:00990106
0:host:no:no:5:node1:0:0:030407082441:030407082441:1:00990106
```

lserrlogbyiogrp

The **lserrlogbyiogrp** command displays a list of the errors and events in the log that are related to I/O groups.

Syntax



Parameters

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise

view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

iogroup_id | iogroup_name
(Optional) Specifies the object ID that filters the log.

Description

This command displays a list of the errors and events in the log that are related to I/O groups. The list can be filtered further by specifying a specific object ID or name. This list displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly, the last *x* entries against a given object type or object ID can be listed.

Note: Although an object type *unknown* is displayed in the error log, there is no available command to filter this object type.

An invocation example

```
svcinfo lserrlogbyiogrp -delim :
```

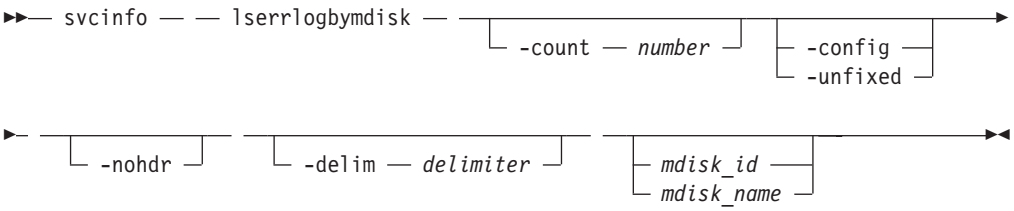
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error_code
1:io_grp:no:no:1:node1:109:109:030407094417:030407094417:1:00000001
```

lserrlogbymdisk

The **lserrlogbymdisk** command displays a list of the errors and events in the log that are related to a specific MDisk.

Syntax



Parameters

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

mdisk_id | *mdisk_name*
(Optional) Specifies the object ID that filters the log.

Description

This command displays a list of the errors and events in the log that are related to a specific MDisks The list can be filtered further by specifying a specific object ID or name. This list displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly, the last *x* entries against a given object type or object ID can be listed.

Note: Although an object type *unknown* is displayed in the error log, there is no available command to filter this object type.

An invocation example

```
svcinfo lserrlogbymdisk -delim :
```

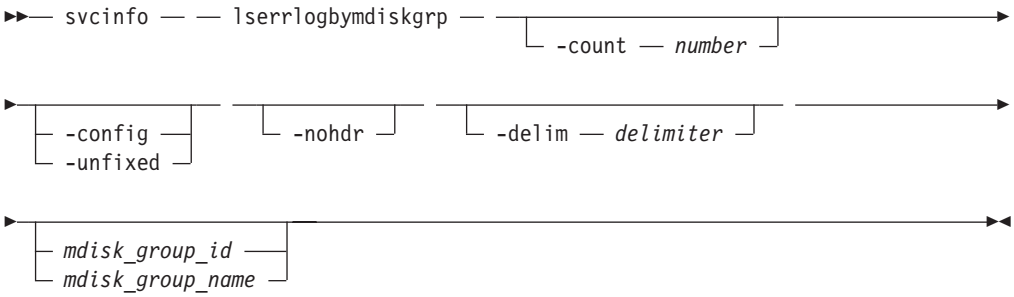
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:
sequence_number:root_sequence_number:first_timestamp:
last_timestamp:number_of_errors:error_code
11:mdisk:no:no:3:node1:108:108:030407092947:030407092947:1:00000016
11:mdisk:no:no:2:node1:107:107:030407092947:030407092947:1:00000016
```

lserrlogbymdiskgrp

The **lserrlogbymdiskgrp** commands display a list of the errors and events in the log that are related to MDisk groups.

Syntax



Parameters

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

mdisk_group_id | mdisk_group_name
(Optional) Specifies the object ID that filters the log.

Description

The **lserrlogbymdiskgrp** command displays a list of the errors and events in the log that are related to MDisk groups. The list can be filtered further by specifying a specific object ID or name. This list displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly, the last *x* entries against a given object type or object ID can be listed.

Note: Although an object type *unknown* is displayed in the error log, there is no available command to filter this object type.

An invocation example

```
svcinfo lserrlogbymdiskgrp -delim :
```

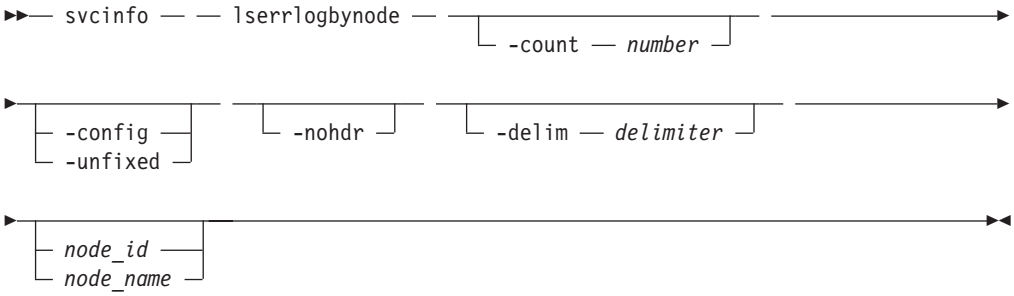
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error_code
1:mdisk_grp:no:no:5:node1:0:0:030407081619:030407081619:1:00990148
128:mdisk_grp:no:no:5:node1:0:0:030407081610:030407081610:1:00990173
0:mdisk_grp:no:no:5:node1:0:0:030407081610:030407081610:1:00990148
```

lserrlogbynode

The **lserrlogbynode** command displays a list of the errors and events in the log that are related to nodes.

Syntax



Parameters

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

node_id | *node_name*
(Optional) Specifies the object ID that filters the log.

Description

This command displays a list of the errors and events in the log that are related to nodes. The list can be filtered further by specifying a specific object ID or name. This list displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly, the last *x* entries against a given object type or object ID can be listed.

Note: Although an object type *unknown* is displayed in the error log, there is no available command to filter this object type.

An invocation example

```
svcinfo lserrlogbynode -delim :
```

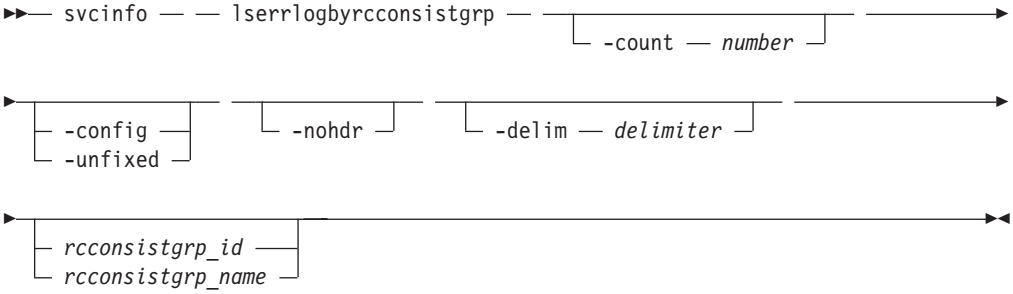
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error_code
1:node:no:no:5:node1:0:0:030407082722:030407082722:1:00990501
1:node:no:no:5:node1:0:0:030407082716:030407082716:1:00990501
1:node:no:no:5:node1:0:0:030407052546:030407052546:1:00990383
0:node:no:no:6:node1:105:105:030407082202:030407082717:2:00980500
1:node:no:no:1:node1:102:102:030407052547:030407052547:1:00074001
```

lserrlogbyrcconsistgrp

You can use the **lserrlogbyrcconsistgrp** command to display the error log by Metro or Global Mirror consistency groups.

Syntax



Parameters

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

rcconsistgrp_id | rcconsistgrp_name
(Optional) Specifies the object ID that filters the log.

Description

This command displays a list of the errors and events in the log by Metro or Global Mirror consistency groups. The list can be filtered further by specifying a specific object ID or name. This displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly, the last *x* entries against a given object type or object ID can be listed.

Note: Although an object type *unknown* is displayed in the error log, there is no available command for this object type.

An invocation example

```
svcinfo lserrlogbyrcconsistgrp -delim :
```

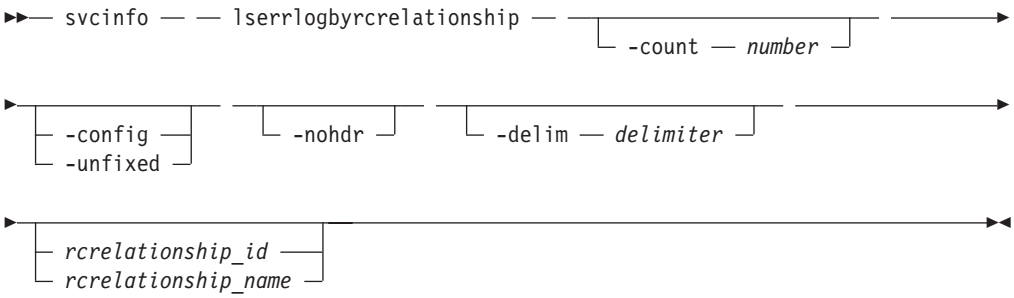
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error_code
253:rc_const_grp:no:no:5:node1:0:0:030407090333:030407090333:1:00990240
254:rc_const_grp:no:no:5:node1:0:0:030407090327:030407090327:1:00990240
255:rc_const_grp:no:no:5:node1:0:0:030407090323:030407090323:1:00990240
```

lserrlogbyrcrelationship

The **lserrlogbyrcrelationship** command displays a list of the errors and events in the log by Metro Mirror or Global Mirror relationships.

Syntax



Parameters

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

rcrelationship_id | rcrelationship_name
(Optional) Specifies the object ID that filters the log.

Description

This command displays a list of the errors and events in the log by Metro or Global Mirror relationships. The list can be filtered further by specifying a specific object ID or name. This displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly the last *x* entries against a given object type or object ID can be listed.

Note: Although an object type *unknown* is displayed in the error log, there is no available command to filter this object type.

An invocation example

```
svcinfo lserrlogbyrcrelationship -delim :
```

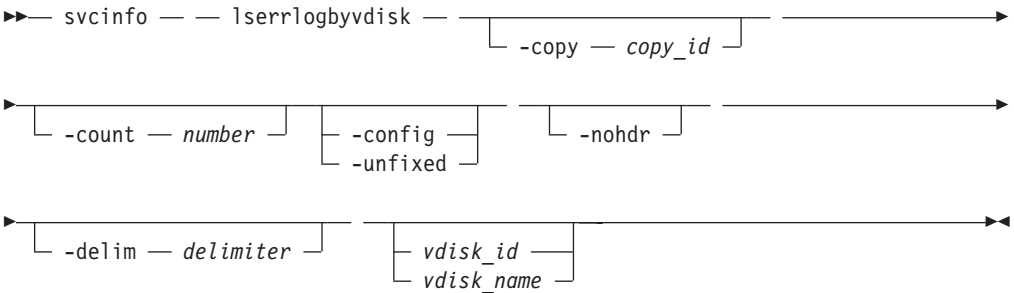
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error_code
2:remote:no:no:5:node1:0:0:030407090442:030407090442:1:00990226
2:remote:no:no:5:node1:0:0:030407090106:030407090106:1:00990225
1:remote:no:no:5:node1:0:0:030407085932:030407085932:1:00990225
2:remote:no:no:6:n/a:106:106:030407090117:030407090117:1:00985002
```

lserrlogbyvdisk

The **lserrlogbyvdisk** command displays a list of the errors and events in the log by VDisks.

Syntax



Parameters

-copy *copy_id*
(Optional) Displays a list of errors and events for the specified VDisk copy.

-count *number*
(Optional) Specifies the maximum number of errors or events to list.

-config
(Optional) Lists only configuration events.

-unfixed
(Optional) Lists only unfixed errors.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

vdisk_id | *vdisk_name*
(Optional) Specifies the object ID that filters the log.

Description

This command displays a list of the errors and events in the log by VDisks. The list can be filtered further by specifying a specific object ID or name. This list displays only the errors and events that have been logged against the specified object. The list can also be filtered to show only the configuration events or the unfixed errors for the given object type or object ID. Similarly, the last *x* entries against a given object type or object ID can be listed.

Note: Although an object type *unknown* is displayed in the error log, there is no available command to filter this object type.

An invocation example

```
svcinfo lserrlogbyvdisk -delim :
```

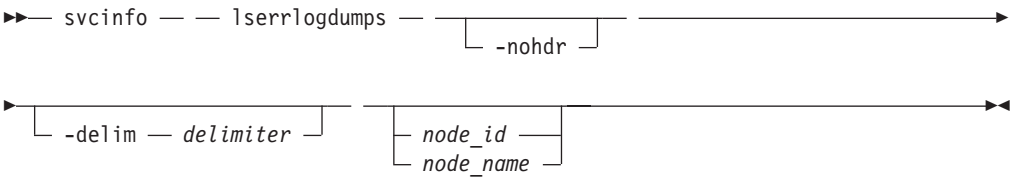
The resulting output

```
id:type:fixed:SNMP_trap_raised:error_type:node_name:type:sequence_number:
root_sequence_number:first_timestamp:last_timestamp:number_of_errors:error
code:copy_id
4:vdisk:no:no:0::vdisk:0:0:021009082703:-300101000000:0:990169:0
1:vdisk:no:no:0::vdisk:0:0:021009081951:-300101000000:0:990182:0
2:vdisk:no:no:0::vdisk:0:0:021009081915:-300101000000:0:990182:0
0:vdisk:no:no:0::vdisk:0:0:021009081835:-300101000000:0:990182:1
```

lserrlogdumps

The **lserrlogdumps** command displays a list of error log dumps in the /dumps/elogs directory. These dumps are created as a result of the **svctask dumperrlog** command.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

node_id | node_name
(Optional) Specifies the node ID or name to list the available dumps of the given type. If you do not specify a node, the dumps that are available on the configuration node are listed.

Description

This command displays a list of error log dumps. These dumps are created as a result of the **svctask dumperrlog** command. An error log dump describes the contents of the error log at the time that the command was issued. If you do not specify a node, the dumps that are available on the configuration node are listed. The command displays files from the **/dumps/elogs** directory.

An invocation example

```
svcinfo lserrlogdumps
```

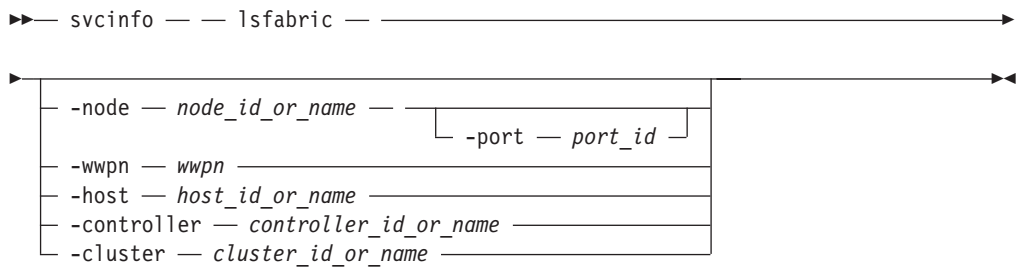
The resulting output

```
id              filename
0               errlog_lynn02_030327_154511
1               aaa.txt_lynn02_030327_154527
2               aaa.txt_lynn02_030327_154559
3               errlog_lynn02_030403_110628
```

lsfabric

The **lsfabric** command generates a report that displays the connectivity between nodes and other controllers and hosts.

Syntax



Parameters

-node *node_id_or_name*
(Optional) Displays the output for all ports for the specified node. The only parameter that you can specify with the **-node** parameter is the **-port** parameter.

-port *port_id*
(Optional) Displays a concise view of all WWPNs that are logged into the specified port ID and node. The **-port** parameter must be specified with only the **-node** parameter. A valid *port_id* value is a number from 1 - 4 that specifies the port number in the vital product data (VPD) or the hexadecimal WWPN of the local port.

-wwpn *wwpn*
(Optional) Displays a list of all ports that have a login to the specified WWPN. You cannot use the **-wwpn** parameter with any other parameter.

-host *host_id_or_name*
(Optional) Specifies a host name or ID. Issuing the **lsfabric** command with the **-host** parameter is equivalent to issuing the **svcinfo lsfabric -wwpn** *wwpn* command for every configured WWPN of the specified host. For example, a host with two ports that are zoned to one port of every node in a eight-node cluster produces 16 lines of output. You cannot use the **-host** parameter with any other parameter.

-controller *controller_id_or_name*
(Optional) Specifies a controller ID or name. You cannot use the **-controller** parameter with any other parameter in this command. Issuing the **lsfabric** command with the **-controller** parameter is equivalent to issuing the **svcinfo lsfabric -wwpn** *wwpn* command for every configured WWPN of the specified controller. For example, a controller with 4 ports connected to a 8 node cluster with 2 counter part SANs produces 64 lines of output.

-cluster *cluster_id_or_name*
(Optional) Specifies a cluster ID or name. You cannot use the **-cluster** parameter with any other parameter. Issuing the **lsfabric** command with the **-cluster** parameter is equivalent to issuing the **svcinfo lsfabric -wwpn** *wwpn* command for every known WWPN in the specified cluster. Output is sorted by remote WWPNs and then cluster WWPNs. This parameter can be used to check the state of connections within the local cluster or between the local and

remote cluster. When the local cluster ID or name is specified, each node-to-node connection is listed twice: once from each end. For example, an eight-node cluster with two counterpart SANs produces eight nodes, multiplied by seven other nodes, multiplied by two SANs, multiplied by four point-to-point logins, equals 448 lines of output.

Description

The **lsfabric** command can be issued with any of the parameters to display a limited subset of information. If the command is issued without any parameters, it provides output for every node.

Values for the **Type** and **State** columns are:

state active
The meaning of this value depends on the object that it applies to, as follows:
- **host or controller**: SCSI commands were issued within the last 5 minutes.
- **node**: node ports can see other ports.

state inactive
No transactions have completed within the last 5 minutes.

Note: It can take up to 10 seconds after a command for a controller port to change from inactive to active. It can take up to 5 minutes after a command for a host port to change from inactive to active.

type One of the following values is displayed:
- **host**
- **node**
- **controller**
- **unknown**

You can issue this command to view all the information about the connections that are available to your cluster.

An invocation example

```
svcinfo lsfabric -delim :
```

The resulting output Each row of output contain the following colon-separated columns:

```
remote_wwpn:remote_nportid: id:node_name:local_wwpn:
local_port:local_nportid:state:name:cluster_name:type
```

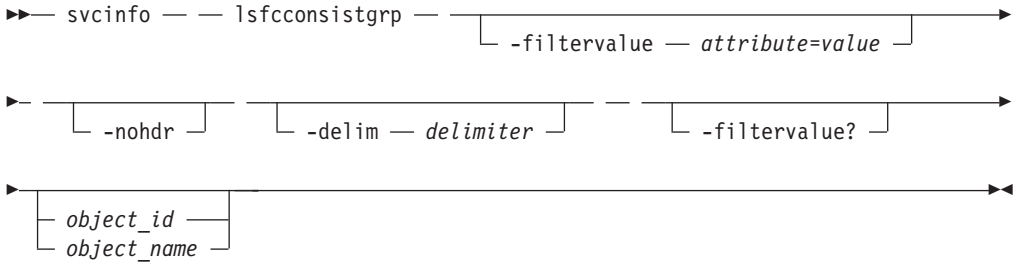
lsfcconsistgrp

The **lsfcconsistgrp** command returns a concise list or a detailed view of FlashCopy consistency groups that are visible to the cluster. This information is useful for tracking FlashCopy consistency groups.

The list report style can be used to obtain two styles of report:
- A list containing concise information about all of the FlashCopy consistency groups on a cluster. (Each entry in the list corresponds to a single FlashCopy consistency group.)

- The detailed information about a single FlashCopy consistency group.

Syntax



Parameters

-filtervalue *attribute=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are returned.

Note: Some filters allow the use of a wildcard when you enter the command. The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk character (*).
- The command can contain a maximum of one wildcard, which must be the first or last character in the string.
- When you use a wildcard, surround the filter entry with double quotation marks (""), as follows:

```
svcinfo lsfcconsistgrp -filtervalue "name=md*"
```

-nohdr
(Optional) By default, headings are displayed for each item of data in a concise view. The **-nohdr** parameter suppresses the display of these headings. Detailed view is not valid for this command.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, the headers are displayed, and the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | *object_name*
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is specified by the **-filtervalue** parameter returns an error message. If you do not specify the *object_id* or *object_name* parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

-filtervalue?
(Optional) Displays the list of valid filter attributes in the report. The valid filter attributes for the **svcinfo lsfcconsistgrp** command are:

- **name**
- **id**
- **status**
- **FC_group_id**

Description

This command returns a concise list or a detailed view of FlashCopy consistency groups that are visible to the cluster.

The following list provides values of the *status* attribute that are displayed as data in the output views:

status **empty**, **idle_or_copied**, **preparing**, **prepared**, **copying**, **stopped**, **suspended**, **stopping**

A concise invocation example

```
svcinfo lsfcconsistgrp -delim :
```

The concise resulting output

```
id:name:status
1:ffccg0:empty
2:ffccg1:idle_or_copied
3:ffccg2:idle_or_copied
```

A detailed invocation example

```
svcinfo lsfcconsistgrp -delim : 1
```

The detailed resulting output

```
id:1
name:ffccg0
status:empty
```

A detailed invocation example

```
svcinfo lsfcconsistgrp -delim : fccstgrp0
```

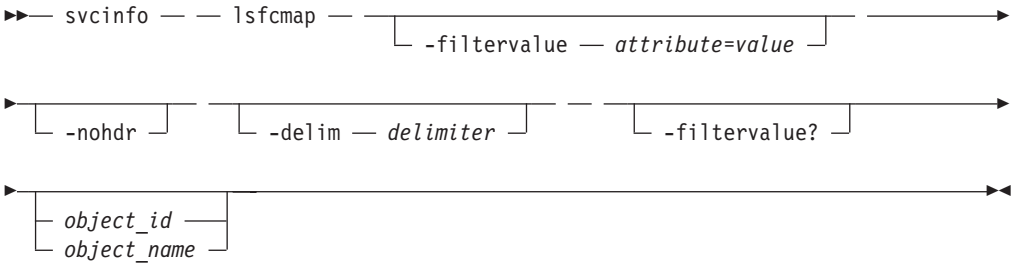
The detailed resulting output

```
id:1
name:FCcgrp0
status:idle_or_copied
autodelete:off
FC_mapping_id:0
FC_mapping_name:fcmap0
FC_mapping_id:1
FC_mapping_name:fcmap1
```

lsfcmap

The **lsfcmap** command generates a list containing concise information about all of the FlashCopy mappings that are visible to the cluster, or detailed information for a single FlashCopy mapping.

Syntax



Parameters

-filtervalue *attribute=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are displayed.

Note: Some filters allow the use of a wildcard when you enter the command. The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is the asterisk (*), which must be used as the first or last character in the string.
- The command can contain a maximum of one wildcard.
- When you use a wildcard, enclose the filter entry within double quotation marks (""), as follows:

```
svcinfo lsfcmap -filtervalue "name=md*"
```

-filtervalue?
(Optional) Displays the valid filter attributes for the **-filtervalue** *attribute=value* parameter:

- name
- id
- source_vdisk_id
- source_vdisk_name
- target_vdisk_id
- target_vdisk_name
- group_name
- group_id
- status
- copy_rate
- FC_mapping_name
- FC_id

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible

width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | object_name
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is specified by the **-filtervalue** parameter is ignored. If you do not specify the **-object_ID** or **object_name** parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

Description

This command returns a concise list or a detailed view of FlashCopy mappings that are visible to the cluster.

The following list shows attribute values that can be displayed as output view data:

status idle_or_copied, preparing, prepared, copying, stopped, suspended or stopping

start_time
Displays the time that the copy was last started. It is in the format *YYMMDDHHMMSS*. If a copy has not been started, a blank line is displayed.

A concise invocation example

```
svcinfo lsfcmap -delim :
```

The concise resulting output

```
id:name:source_vdisk_id:source_vdisk_name:target_vdisk_id:target_vdisk_name:
group_id:group_name:status:progress:copy_rate:clean_progress:incremental
0:fcmap0:63:vdisk63:57:vdisk57:::idle_or_copied:0:0:100:off
1:fcmap1:64:vdisk64:58:vdisk58:::idle_or_copied:0:0:100:on
```

A detailed invocation example

```
svcinfo lsfcmap -delim : 0
```

The detailed resulting output

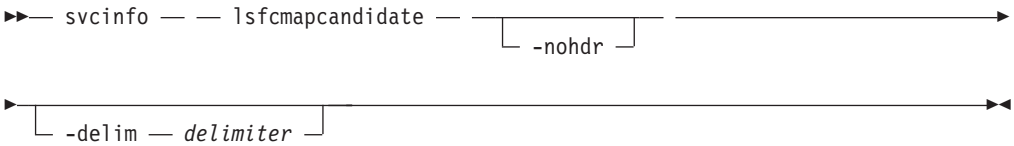
```
id:0
name:fcmap0
source_vdisk_id:63
source_vdisk_name:vdisk63
target_vdisk_id:57
target_vdisk_name:vdisk57
group_id:
group_name:
status:idle_or_copied
progress:0
copy_rate:0
start_time:
dependent_mappings:0
autodelete:off
clean_progress:100
```

```
clean_rate:50
incremental:off
difference:100
grain_size:256
IO_group_id:1
IO_group_name:io_grp1
```

lsfcmapcandidate

The **lsfcmapcandidate** command lists all of the VDisks that are associated with fewer than 256 FlashCopy mappings.

Syntax



Parameters

-nohdr
(Optional) By default, the heading is displayed for the column of data in a concise style view, and for the item of data in a detailed style view. The **-nohdr** parameter suppresses the display of the heading.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter **-delim :** on the command line, a colon character (**:**) separates all items of data in a concise view; the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command returns a list of VDisks that are associated with fewer than 256 FlashCopy mappings.

An invocation example

```
svcinfo lsfcmapcandidate
```

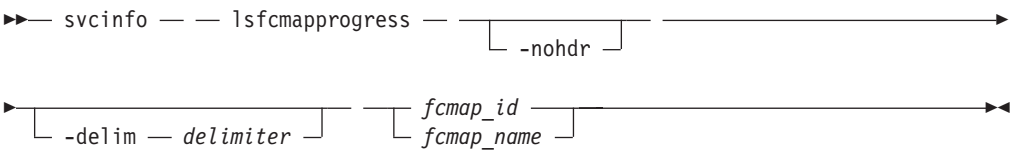
The resulting output

```
id
2
3
4
```

lsfcmapprogress

The **lsfcmapprogress** command returns the progress of the background copy of a FlashCopy mapping. This is displayed as a percentage completed value.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the data is separated from its header by a colon character (**:**).

fcmap_id | *fcmap_name*
(Required) Specifies that you want the report to display the progress of the background copy for the designated FlashCopy mapping.

Description

This command reports a percentage for the progress of the background copy being done on the specified FlashCopy mapping.

An invocation example

```
svcinfo lsfcmapprogress 0
```

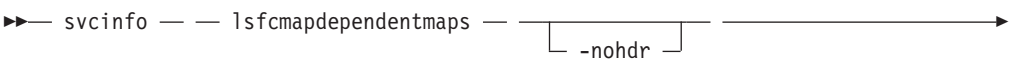
The resulting output

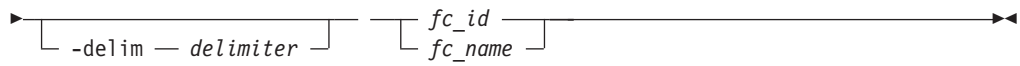
```
id              progress
0               0
```

lsfcmapdependentmaps

The **lsfcmapdependentmaps** command displays all the FlashCopy mappings that are dependent on the user specified mapping.

Syntax





Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

fc_id | fc_name
(Required) Specifies the name or ID of the FlashCopy mapping to list the dependent maps for.

Description

This command returns a list of dependent FlashCopy mappings. This command can be used to determine the list of FlashCopy mappings that would also stop if you stopped a mapping using the **-force** parmeter.

There is a dependent_mapping_count field in the FlashCopy map detailed view (displayed when you process the **lsfcmap** command) that you can use as an indicator as to whether there are any dependent mappings in progress. If the count is zero, there are no dependent copies.

Note: If a period time elapses between the time you process the **lsfcmap** command and the **lsfcmapdependentmaps** command, there could be a difference between the actual number of dependent mappings being processed and the number that was reported by the **lsfcmap** command.

A concise invocation example

```
svcinfo lsfcmapdependentmaps -delim : 2
```

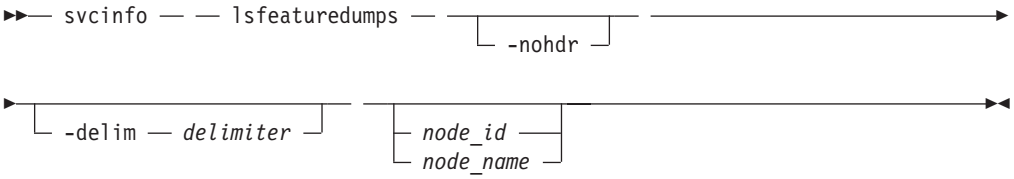
The resulting output

```
fc_id:fc_name
1:fcmap1
3:fcmap3
```

lsfeaturedumps

The **lsfeaturedumps** command displays a list of dump files in the **/dumps/feature** directory. These dump files are created as a result of the **svctask dumpinternallog** command.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

node_id | *node_name*
(Optional) Specifies the node ID or node name to list the available dumps of the given type. If you do not specify a node, the dumps that are available on the configuration node are listed.

Description

This command displays a list of featurization dumps. These dumps are created as a result of issuing the **svctask dumpinternallog** command. A featurization dump file describes the contents of the featurization log at the time that the command was issued. If you do not specify a node, the dumps that are available on the configuration node are listed. The command displays files from the **/dumps/feature** directory.

Issue the **svcinfo lsfeaturedumps** command to display a list of dumps in the **/dumps/feature** destination directory. The feature log is maintained by the cluster. The feature log records events that are generated when license parameters are entered or when the current license settings have been breached.

An invocation example

```
svcinfo lsfeaturedumps
```

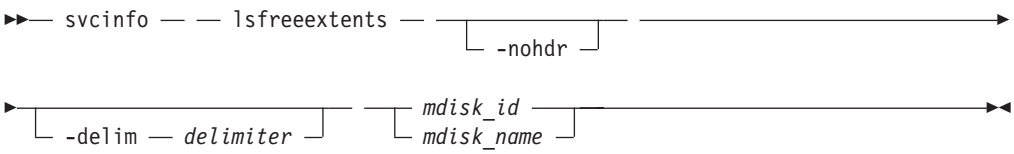
The resulting output

```
id              feature_filename
0               feature.txt
```

lsfreeextents

The **lsfreeextents** command lists the number of free extents that are available on a specified MDisk.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

mdisk_id | *mdisk_name*
(Required) Specifies the ID or the name of the MDisk for which you want to know the number of free extents.

Description

This command displays a count of the number of free extents on the specified MDisk.

An invocation example

```
svcinfo lsfreeextents 2
```

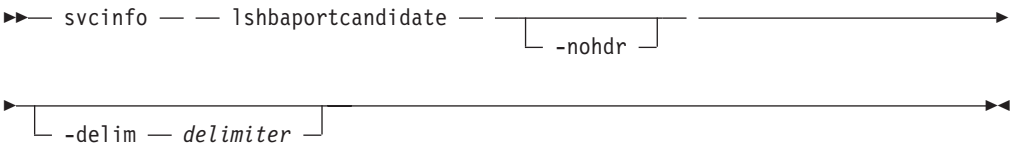
The resulting output

```
id 2
number_of_extents 4372
```

lshbaportcandidate

The **lshbaportcandidate** command lists all of the unconfigured, logged-in host bus adapter (HBA) ports. This information is used to find open HBA ports.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command returns a list of unconfigured, logged in HBA ports.

Note: The **svcinfo lshbaportcandidate** command presents a list of host HBA ports that are logged in to nodes. However, there are situations when the information that is presented might include host HBA ports that are no longer logged in or even part of the SAN fabric. For example, a host HBA port is unplugged from a switch but **svcinfo lshbaportcandidate** still shows the WWPN that is logged in to all nodes. If this occurs, the incorrect entry is removed when another device is plugged in to the same switch port that previously contained the removed host HBA port.

An invocation example

```
svcinfo lshbaportcandidate
```

The resulting output

```
id
210100E08B2520D4
```

lshost

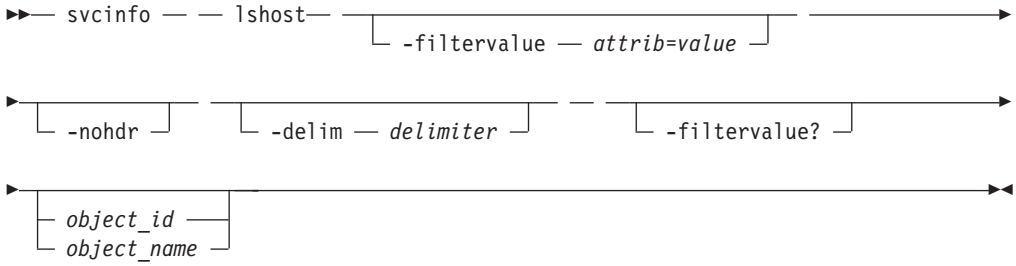
The **lshost** command generates a list with concise information about all the hosts visible to the cluster and detailed information about a single host.

The list report style can be used to obtain two styles of report:

- A list containing concise information about all the hosts visible to the cluster. Each entry in the list corresponds to a single host.

- The detailed information about a single host.

Syntax



Parameters

-filtervalue *attrib=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are returned. If a capacity is specified, the units must also be included.

Note: Some filters allow the use of a wildcard when you enter the command. The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard.
- When using a wildcard character, you must enclose the filter entry within double quotation marks (" "), as follows:

```
svcinfo lshost -filtervalue  "name=md*"
```

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | object_name
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id | object_name* parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

-filtervalue?

(Optional) Specifies that you want your report to display any or all of the list of valid filter attributes. The valid filter attributes for the **svcinfo lshost** command are:

- host_name
- host_id
- port_count
- name
- id
- iogrp_count

Description

This command returns a concise list or a detailed view, of hosts visible to the cluster.

The following list provides the different states for a host:

offline
The host is offline if one or more I/O groups with VDisk mappings does not have a login for the specified WWPN.

degraded
The host is degraded if one or more nodes with VDisk mappings do not have a login for the specified WWPN.

inactive
The host is inactive if all the nodes with VDisk mappings have a login for the specified WWPN, however, no nodes have seen any SCSI commands from the WWPN in the last 5 minutes.

active The host is active if all the nodes with VDisk mappings have a login for the specified WWPN, however, at least one node has seen SCSI commands from the WWPN in the last 5 minutes.

If a host does not have any VDisk mappings, then it is either reported as offline or inactive.

Note: The **svcinfo lshost** command presents a list of host HBA ports that are logged in to nodes. However, there are situations when the information presented can include host HBA ports which are no longer logged in or even part of the SAN fabric. For example: A host HBA port is unplugged from a switch but **scvinfo lshost** still shows the WWPN logged in to all nodes. If this occurs, the incorrect entry is removed when another device is plugged in to the same switch port that previously contained the removed host HBA port.

A concise invocation example

```
svcinfo lshost -delim :
```

The concise resulting output

```
id:name:port_count:iogrp_count
0:host0:1:0
1:host1:1:0
```

A detailed invocation example

```
svcinfo lshost -delim : 1
```

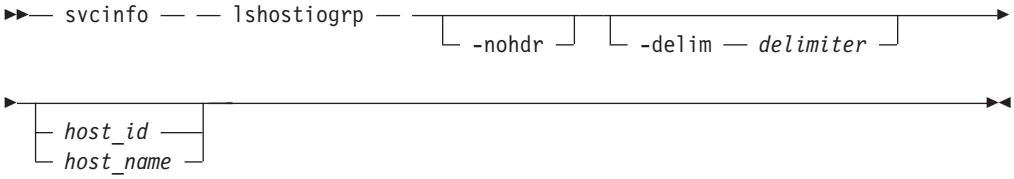
The detailed resulting output

```
id:1
name:host1
port_count:1
type:generic
iogrp_count:0
mask:1111
WWPN:000000000001AABB
node_logged_in_count:0
state:inactive
```

lshostiogrp

The **lshostiogrp** command displays a list of all the I/O groups that are associated with a specified host.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

host_id | *host_name*
(Required) The name or ID of the host for which the list of I/O groups is required.

Description

This command displays a list of all the I/O groups that are mapped to the specified host.

An invocation example

```
svcinfo lshostiogrp -delim : hostone
```

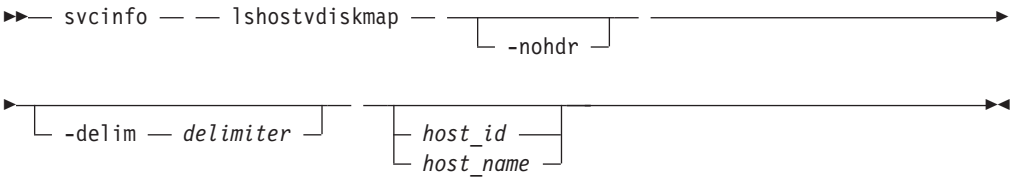
The resulting output

```
id:name
0:io_grp0
1:io_grp1
```

lshostvdiskmap

The **lshostvdiskmap** command displays a list of VDisks that are mapped to a given host. These are the VDisks that are recognized by the specified host.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

host_id | *host_name*
(Optional) Specifies the host in terms of its ID or name. The command displays a list of all the virtual disks that are mapped to the specified host and the SCSI ID by which they are mapped. If neither a host ID or name are entered, the command displays a list of all recognized host-to-VDisk mappings.

Description

This command displays a list of VDisk IDs and names. These are the VDisks that have been mapped to the specified host; that is, they are visible to the specified host. The SCSI LUN ID is also displayed. This SCSI LUN ID is the ID by which the VDisk is recognized by the host.

Each VDisk that is exported by the cluster is assigned a unique vpath number. This number identifies the VDisk and determines which VDisk corresponds to the volume that the hosts recognize. This procedure can only be performed using the command-line interface.

For the specified volume, find the vpath serial number by issuing the **datapath query device** command. Find the host that is defined to the cluster that corresponds with the host that you are working with.

1. The WWPNs are an attribute of the HBA. You can find these by looking at the device definitions stored by your operating system. For example, on AIX they are in the ODM, in Windows they are in the Device Manager details for the given HBA.
2. Verify which host is defined to the cluster that these ports belong to. The ports are stored as part of the detailed view, so you must list each host in turn by issuing the following command:

   ```
   svcinfo lshost host_name | host_id
   ```

   where *host_name | host_id* is the name or ID of the host. Check for matching WWPNs.

   Note: Name your hosts accordingly. For example, if the actual host is called **orange**, also name the host that is defined to the cluster **orange**.

When you have the *hostname* defined to the cluster and the *vpath serial number*, issue the following command:

```
svcinfo lshostvdiskmap hostname
```

where *hostname* is the name of the host. A list is displayed. Look for the VDisk UID that matches the *vpath serial number* and record the VDisk name or ID.

An invocation example

```
svcinfo lshostvdiskmap -delim : 2
```

The resulting output

```
id:name:SCSI_id:vdisk_id:vdisk_name:wwpn:vdisk_UID
2:host2:0:10:vdisk10:0000000000000ACA:600507680195800150000000000000000A
2:host2:1:11:vdisk11:0000000000000ACA:600507680195800150000000000000000B
2:host2:2:12:vdisk12:0000000000000ACA:600507680195800150000000000000000C
2:host2:3:13:vdisk13:0000000000000ACA:600507680195800150000000000000000D
2:host2:4:14:vdisk14:0000000000000ACA:600507680195800150000000000000000E
```

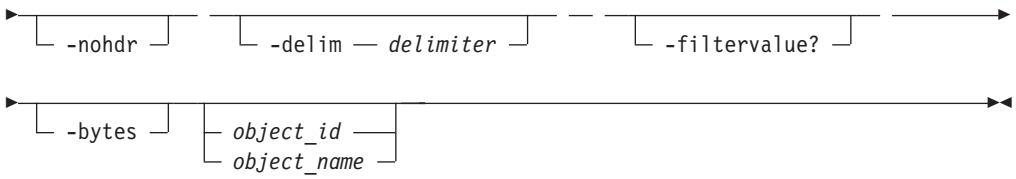
lsiogrp

The **lsiogrp** command returns a concise list or a detailed view of I/O groups visible to the cluster.

The list report style can be used to obtain the following two styles of report:

- A list containing concise information about all the I/O groups that are visible to the cluster. Each entry in the list corresponds to a single I/O group.
- The detailed information about a single I/O group.

Syntax



Parameters

-filtervalue *attrib=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are returned. If a capacity is specified, the units must also be included.

Note: Some filters allow the use of a wildcard when you enter the command. The following rules apply to the use of wildcard characters with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*), which must be the first or last character in the string.
- The command can contain a maximum of one wildcard.
- When you use a wildcard, enclose the filter entry within double quotation marks (""), as follows:

```
svcinfo lsiogrp -filtervalue "name=md*"
```

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-filtervalue?
(Optional) Displays the valid filter attributes for the **svcinfo lsiogrp** command:

- **HWS_name**
- **HWS_unique_id**
- **node_count**
- **vdisk_count**
- **name**
- **id**
- **host_count**

-bytes
(Optional) Displays all capacities as bytes.

object_id | *object_name*
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is

specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id | object_name* parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

Description

This command returns a concise list or a detailed view of I/O groups visible to the cluster.

A concise invocation example

```
svcinfo lsiogrp -delim :
```

The concise resulting output

```
id:name:node_count:vdisk_count:host_count
0:io_grp0:1:0:0
1:io_grp1:0:0:0
2:io_grp2:0:0:0
3:io_grp3:0:0:0
4:recovery_io_grp:0:0:0
```

A detailed invocation example

```
svcinfo lsiogrp -delim : 0
```

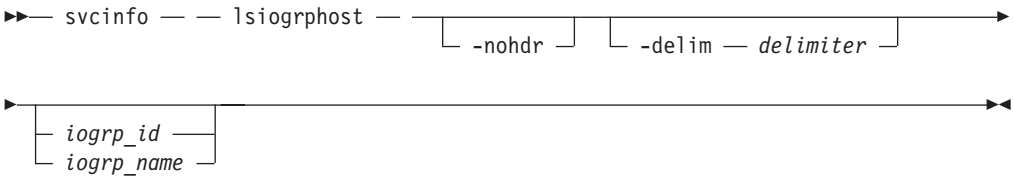
The detailed resulting output

```
id:0
name:io_grp0
node_count:1
vdisk_count:51
host_count:0
flash_copy_total_memory:3.0MB
flash_copy_free_memory:1.0MB
remote_copy_total_memory:6.5MB
remote_copy_free_memory:2.8MB
mirroring_total_memory:1.0MB
mirroring_free_memory:0.3MB
```

lsiogrphost

The **lsiogrphost** command displays a list of the hosts that are mapped to a specified I/O group.

Syntax



Parameters

-nohdr

(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

iogrp_id | iogrp name
(Required) The ID or name of the I/O group for which a list of all mapped hosts is required.

Description

The **lsiogrphost** command displays a list of hosts that are mapped to a specified I/O group.

An invocation example

```
svcinfo lsiogrphost -delim : 0
```

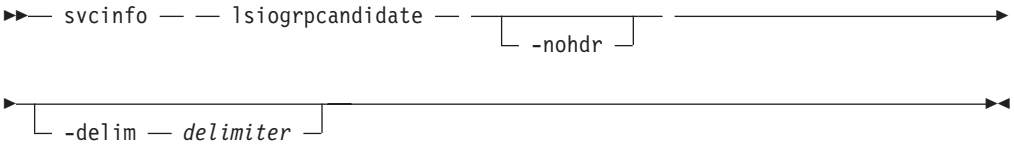
The resulting output

```
id:name
0:hostzero
1:hostone
```

lsiogrpcandidate

Use the **lsiogrpcandidate** command to list the I/O groups that can have nodes added to them.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by

a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command displays a list of I/O groups to which nodes can be added. Only the I/O group IDs are displayed.

An invocation example

```
svcinfo lsiogrpcandidate
```

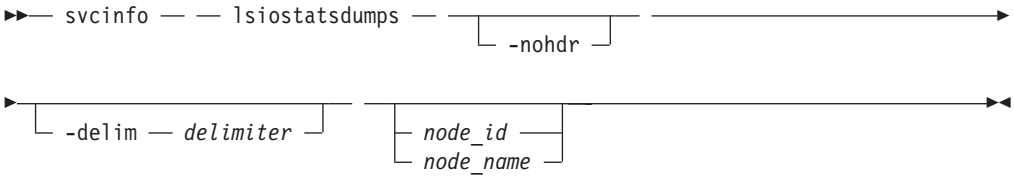
The resulting output

```
id
0
1
2
3
4
```

lsiostatsdumps

The **lsiostatsdumps** command displays a list of dumps in the **/dumps/iostats** directory. Use the **svctask startstats** command to create these dumps.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

node_id | node_name
(Optional) Specifies the node ID or name to list the available dumps of the given type. If you do not specify a node, the dumps that are available on the configuration node are listed.

Description

This command displays a list of I/O statistics dumps. These dumps are created when you issue the **svctask startstats** command. If you do not specify a node, the dumps that are available on the configuration node are listed. The command displays files from the /dumps/iostats directory.

An invocation example

```
svcinfo lsiostatsdumps
```

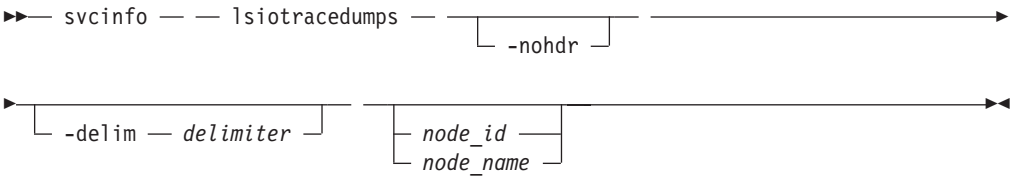
The resulting output

```
id               iostat_filename
0                v_stats_mala75_031123_072426
1                m_stats_mala75_031123_072425
```

lsiotracedumps

Use the **lsiotracedumps** command to display a list of files in the **/dumps/iotrace** directory.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

node_id | node_name
(Optional) Specifies the node ID or name to list the available dumps of the given type. If you do not specify a node, the dumps that are available on the configuration node are listed.

Description

This command displays a list of I/O trace dumps. These dumps are created when you issue the **svctask settrace** command. If you do not specify a node, the dumps that are available on the configuration node are listed. The command displays files from the **/dumps/iotrace** directory.

An invocation example

```
svcinfo lsiotracedumps
```

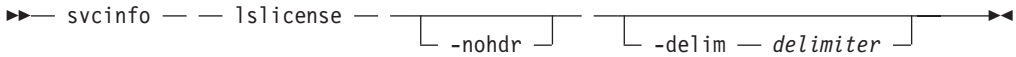
The resulting output

```
id               iotrace_filename
0                c1_mala75_030405_092155
1                c2_mala75_030405_092156
2                c3_mala75_030405_092158
3                c4_mala75_030405_092159
4                c5_mala75_030405_092201
```

lslicense

The **lslicense** command displays current license settings for cluster features.

Syntax



Parameters

-nohdr
(Optional) Suppresses the display of these headings. By default, headings are displayed for each column of data (in a concise style view providing general information about objects of a particular type) and for each item of data (in a detailed style view providing much more information about a specific object of a particular type).

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :`, a colon character (:) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

The **lslicense** command displays license settings for cluster features, including FlashCopy, RemoteCopy, and Virtualization settings. The displayed output lists feature enablement and capacities.

Use the **chlicense** command to change the feature license settings. Because the feature license settings are entered when the cluster is first created, you must only update the settings if you have changed your license.

An invocation example

```
svcinfo lslicense
```

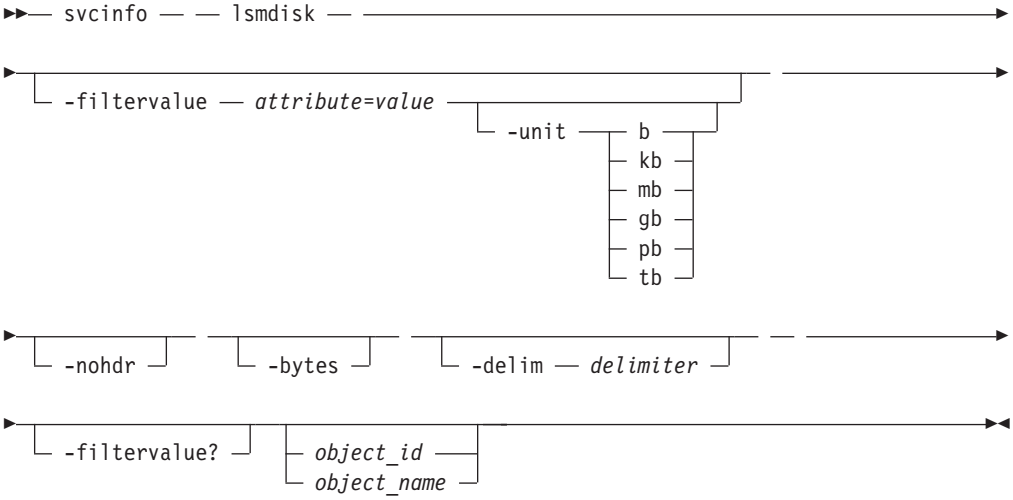
The resulting output

```
used_flash 4.73
used_remote 0
used_virtualization 21.12
license_flash 5
license_remote 0
license_virtualization 32
license_physical_disks 0
license_physical_flash off
license_physical_remote off
```

lsmdisk

The **lsmdisk** command returns a concise list or a detailed view of MDisks visible to the cluster. It can also list detailed information about a single managed disk.

Syntax



Parameters

-filtervalue *attribute=value*
(Optional) Specifies a list of one or more filter attributes matching the specified values; see **-filtervalue?** for the supported attributes. Only objects with a value that matches the filter attribute value are returned. If **capacity** is specified, the units must also be included.

Note: Some filters allow the use of a wildcard when entering the command. The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard, which must be the first or last character in the string.
- When using a wildcard character, you must enclose the filter entry within double quotation marks (""), as follows:

```
svcinfo lsmdisk -filtervalue  "name=md*"
```

-filtervalue?

(Optional) Includes all of the valid filter attributes in the report. The following filter attributes are valid for the **svcinfo lsmdisk** command:

- name
- id
- status
- mode
- mdisk_grp_id
- mdisk_grp_name
- capacity
- controller_name

Any parameters specified with the **-filtervalue?** parameter are ignored.

-nohdr

(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-bytes

(Optional) Specifies that you want the report to display all capacities as bytes.

-delim *delimiter*

(Optional) By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | *object_name*

(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is specified by the **-filtervalue** parameter is ignored. If you do not specify the **-object_ID** or **-object_name** parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

Description

This command returns a concise list or a detailed view of MDisks visible to the cluster. Table 13 provides the potential output for MDisks.:

Table 13. MDisk output. Describes MDisk output attribute values

Attribute	Values
status	offline, excluded, degraded, online
mode	unmanaged, managed, image
quorum index	0,1,2 valid Quorum index

When back-end controllers are added to the fibre-channel SAN and are included in the same switch zone as a cluster, the cluster automatically discovers the back-end controller. The cluster determines from the controller which type of storage that it is presenting to the node. The SCSI LUs presented by the back-end controller are displayed as unmanaged MDisks. However, if the configuration of the back-end controller is modified after this has occurred, the cluster might be unaware of these configuration changes. This command allows you to re-scan the fibre-channel network to update the list of unmanaged MDisks.

Note: The automatic discovery performed by the cluster does not write anything to an unmanaged MDisk. It is only when you add an MDisk to an MDisk group, or use an MDisk to create an image mode virtual disk, that the system uses the storage.

To see which MDisks are available, issue the **svctask detectmdisk** command to manually scan the fibre-channel network for any MDisks. Issue the **svcinfo lsmdiskcandidate** command to show the unmanaged MDisks. These MDisks have not been assigned to an MDisk group. Alternatively, you can issue the **svcinfo lsmdisk** command to view all of the MDisks.

Each MDisk corresponds with a single RAID array, or a single partition on a given RAID array. Each RAID controller defines a LUN number for this disk. The LUN number and controller name or ID are needed to determine the relationship between MDisks and RAID arrays or partitions.

To determine the relationship between MDisks and RAID arrays or LUNs, enter the following command:

```
svcinfo lsmdisk mdiskname
```

where *mdiskname* is the name of the MDisk.

Note: Record the controller name or controller ID and controller LUN number for later use.

Show the detailed view of the controller by issuing the following command:

```
svcinfo lscontroller controllername
```

where *controllername* is the name of the controller.

Note: Record the vendor ID, product ID, and worldwide node name. Use these to determine what is being presented to the MDisk.

From the native user interface for the given controller, list the LUNs that the

controller is presenting and match the LUN number. This value shows you the exact RAID array or partition that corresponds with the MDisk.

A concise invocation example

```
svcinfo lsmdisk -delim :
```

The concise resulting output

```
id:name:status:mode:mdisk_grp_id:mdisk_grp_name:capacity:
ctrl_LUN_#:controller_name:UID
0:mdisk0:online:managed:0:mdiskgrp0:68.4GB:0000000000000000:controller0:
20000004cf2422aa0000000000000000000000000000000000000000000000000000
1:mdisk1:online:managed:0:mdiskgrp0:68.4GB:0000000000000000:controller1:
20000004cf1fd19d0000000000000000000000000000000000000000000000000000
2:mdisk2:online:managed:0:mdiskgrp0:68.4GB:0000000000000000:controller2:
20000004cf2425310000000000000000000000000000000000000000000000000000
3:mdisk3:online:managed:0:mdiskgrp0:68.4GB:0000000000000000:controller3:
20000004cf1fd7a0000000000000000000000000000000000000000000000000000
4:mdisk4:online:unmanaged:::68.4GB:0000000000000000:controller4:
20000004cf2413080000000000000000000000000000000000000000000000000000
5:mdisk5:online:unmanaged:::68.4GB:0000000000000000:controller5:
20000004cf2412ca0000000000000000000000000000000000000000000000000000
6:mdisk6:online:unmanaged:::68.4GB:0000000000000000:controller6:
20000004cf2421230000000000000000000000000000000000000000000000000000
7:mdisk7:online:unmanaged:::68.4GB:0000000000000000:controller7:
20000004cf2412ac0000000000000000000000000000000000000000000000000000
8:mdisk8:online:unmanaged:::68.4GB:0000000000000000:controller8:
20000004cf2421de0000000000000000000000000000000000000000000000000000
9:mdisk9:online:unmanaged:::68.4GB:0000000000000000:controller9:
20000004cf1fd7cc0000000000000000000000000000000000000000000000000000
10:mdisk10:online:unmanaged:::68.4GB:0000000000000000:controller10:
20000004cf1ffe880000000000000000000000000000000000000000000000000000
11:mdisk11:online:unmanaged:::68.4GB:0000000000000000:controller11:
20000004cf2412aa0000000000000000000000000000000000000000000000000000
12:mdisk12:online:unmanaged:::68.4GB:0000000000000000:controller12:
20000004cf2423740000000000000000000000000000000000000000000000000000
13:mdisk13:online:unmanaged:::68.4GB:0000000000000000:controller13:
20000004cf2419fe0000000000000000000000000000000000000000000000000000
14:mdisk14:online:unmanaged:::68.4GB:0000000000000000:controller14:
20000004cf2419320000000000000000000000000000000000000000000000000000
```

A detailed invocation example

```
svcinfo lsmdisk -delim : 2
```

The detailed resulting output

```
id:2
name:mdisk2
status:online
mode:unmanaged
mdisk_grp_id:
mdisk_grp_name:
capacity:68.4GB
quorum_index:
block_size:512
controller_name:controller2
ctrl_type:4
ctrl_WWNN:20000004CF242531
controller_id:2
path_count:1
max_path_count:1
ctrl_LUN_#:0000000000000000
UID:20000004cf2425310000000000000000000000000000000000000000000000000000
preferred_WWPN:22000004CF242531
active_WWPN:22000004CF242531
```

The following define the ctrl_type fields:

0 Device that has not yet been processed.

1 Device is a node in which that cluster has not yet been identified.

2 Device is a node within this cluster.

3 Device is a node within a remote cluster.

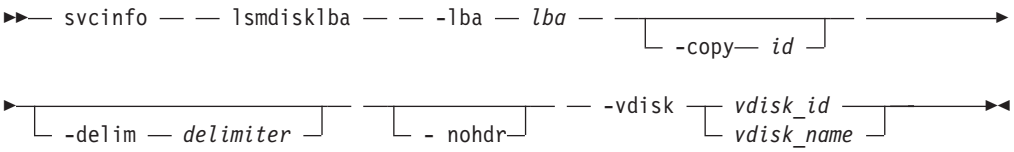
4 Device is a back-end storage controller.

5 Device is a storage or other fabric device that is incompatible with the cluster.

lsmdisklba

The **lsmdisklba** command lists the MDisk and logical block address (LBA) for the specified VDisk LBA.

Syntax



Parameters

-lba *lba*
(Required) Specifies the 64–bit hexadecimal logical block address (LBA) on the VDisk.

-copy *id*
(Optional) Specifies the VDisk copy ID to list the MDisk and LBA for. If this parameter is not specified, the command lists MDisks and LBAs for all VDisk copies.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

vdisk_id | *vdisk_name*
(Required) Specifies the VDisk name or ID.

Description

The **lsmdisklba** command returns one line of output for each copy of the specified VDisk. Each line describes the logical block address (LBA) of the MDisk that is associated with the VDisk LBA. For mirrored VDisks, the command lists the MDisk LBA for both the primary and the copy.

If applicable, the command also lists the range of LBAs on both the VDisk and MDisk that are mapped in the same extent, or for space-efficient disks, in the same grain. If a space-efficient VDisk is offline and the specified LBA is not allocated, the command displays the VDisk LBA range only.

Table 14 summarizes the data that can be returned with this command.

Table 14. lsmdisklba command output. Describes the data that can be returned

Field	Fully allocated, single copy VDisk	LBA not allocated on space-efficient VDisk	Mirrored VDisk with one normal copy and one offline space-efficient copy	Mirrored VDisk with one normal copy and one offline space-efficient copy
copy_id	yes	yes	yes	yes
mdisk_id	yes	no	yes	no
mdisk_name	yes	no	yes	no
type	allocated	unallocated	allocated	offline
lba	yes	no	yes	no
mdisk_start	yes	no	yes	no
mdisk_end	yes	no	yes	no
vdisk_start	yes	yes	yes	yes
vdisk_end	yes	yes	yes	yes

An invocation example

```
svcinfo lsmdisklba -vdisk 32 -lba 0x3480200 -delim :
```

The resulting output

```
id:mdisk_id:mdisk_name:type:lba:mdisk_start:mdisk_end:vdisk_start:vdisk_end
0:32:mdisk32:allocated:0x3480200:0x3480000:0x34808ff:0x42954000:0x429548ff
1:::offline:::0x42954000:0x42954020
```

An invocation example

```
svcinfo lsmdisklba -vdisk 20 -lba 0x9FFFF -delim :
```

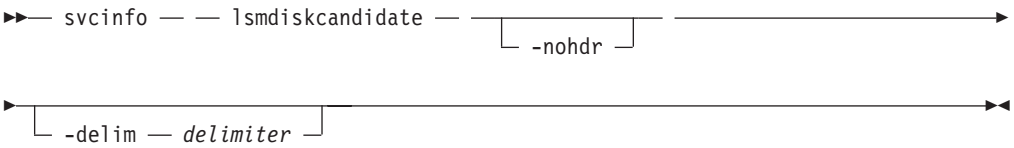
The resulting output

```
copy_id:mdisk_id:mdisk_name:type:LBA:mdisk_start:mdisk_end:vdisk_start:vdisk_end
0:7:mdisk7:allocated:0x003A04FF:0x003A0400:0x003A04FF:0x0009FF00:0x0009FFFF
1:18:mdisk18:allocated:0x0019FFFF:0x00180000:0x001BFFFF:0x00080000:0x000BFFFF
```

lsmdiskcandidate

The **lsmdiskcandidate** command lists all of the unmanaged MDisks by MDisk ID.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command displays a list of MDisks that are unmanaged. Only the MDisk IDs are displayed.

When back-end controllers are added to the fibre-channel SAN and are included in the same switch zone as a cluster, the cluster automatically detects the back-end controller to determine which storage is presented to the node. The SCSI logical units that are presented by the back-end controller are displayed as unmanaged MDisks. However, if the configuration of the back-end controller is modified after this has occurred, the cluster might be unaware of these configuration changes. You can then request that the cluster rescan the fibre-channel SAN to update the list of unmanaged MDisks.

Note: The automatic detection performed by the cluster does not write anything to a unmanaged MDisk. It is only when you instruct the cluster to add an MDisk to a managed disk group or use a MDisk to create an image mode virtual disk that the storage is actually used.

Check to see which MDisks are available by issuing the **svctask detectmdisk** command to manually scan the fibre-channel network for any MDisks. Issue the **svcinfo lsmdiskcandidate** command to show the unmanaged MDisks. These MDisks have not been assigned to an MDisk group. Alternatively, you can issue the **svcinfo lsmdisk** command to view all of the MDisks.

An invocation example

```
svcinfo lsmdiskcandidate
```

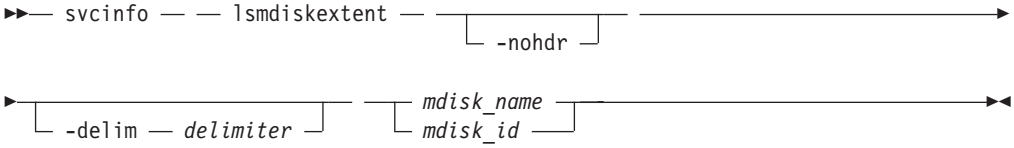
The resulting output

```
id
5
6
7
8
9
10
11
12
13
14
```

lsmdiskextent

The **lsmdiskextent** command displays the extent allocation between managed disks and virtual disks. The output lists a VDisk ID, VDisk copy ID, and the number of extents.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

mdisk_name | *mdisk_id*
(Required) Specifies the specific object ID or name of the given type.

Description

The command displays a list, in which each entry contains a VDisk ID, VDisk copy ID, and the number of extents. These VDisk copies are using extents on the specified MDisk. The number of extents being used on each MDisk is also shown.

Every VDisk copy is constructed from one or more MDisks. At times, you might have to determine the relationship between the two objects. The following procedure allows you to determine the relationships.

To determine the relationship between VDisk copies and MDisks, issue the following command for each VDisk copy:

```
svcinfo lsvdiskmember vdisk_name | vdisk_id
```

where *vdisk_name | vdisk_id* is the name or ID of the VDisk copy. This displays a list of IDs that correspond to the MDisks that make up the VDisk copy.

To determine the relationship between VDisk copies and MDisks and the number of extents that are provided by each MDisk, you must use the command-line interface. For each VDisk copy, issue the following command:

```
svcinfo lsvdiskextent vdisk_name | vdisk_id
```

where *vdisk_name | vdisk_id* is the name or ID of the VDisk copy. This displays a table of MDisk IDs and the corresponding number of extents that each MDisk is providing as storage for the given VDisk copy.

To determine the relationship between MDisks and VDisk copies, issue the following command for each MDisk:

```
svcinfo lsmdiskmember mdisk_name | mdisk_id
```

where *mdisk_name | mdisk_id* is the name or ID of the MDisk. This displays a list of IDs that correspond to the VDisk copies that are using this MDisk.

To determine the relationship between MDisks and VDisk copies and the number of extents that are used by each VDisk copy, you must use the command-line interface. For each MDisk, issue the following command:

```
svcinfo lsmdiskextent mdisk_name | mdisk_id
```

where *mdisk_name | mdisk_id* is the name or ID of the MDisk. This command displays a table of VDisk copy IDs and the corresponding number of extents that are being used by each VDisk copy.

An invocation example

```
svcinfo lsmdiskextent -delim : mdisk0
```

The resulting output

```
id:number_of_extents:copy_id
1:1:1
```

lsmdiskgrp

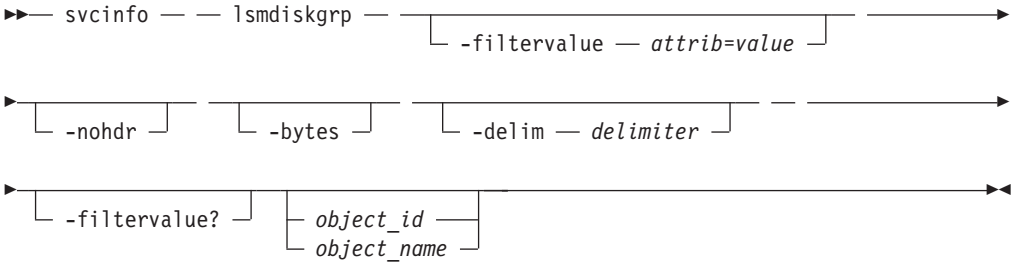
The **lsmdiskgrp** command returns a concise list or a detailed view of MDisk groups visible to the cluster.

The list report style can be used to obtain two styles of report:

- A list containing concise information about all the managed disk groups in a cluster. (Each entry in the list corresponds to a single managed disk group.)

- The detailed information about a single managed disk group.

Syntax



Parameters

-filtervalue *attrib=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are returned. If a capacity is specified, the units must also be included.

Note: Some filters allow the use of a wildcard when you enter the command. The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard, which must be the first or last character in the string.
- When using a wildcard, you must enclose the filter entry within double quotation marks (""), as follows:

```
svcinfo lsmdiskgrp -filtervalue  "name=md*"
```

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-bytes
(Optional) Specifies that you want the report to display all capacities as bytes.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | object_name
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id |*

object_name parameter, the concise view of all objects matching the filtering requirements specified by the **-filtervalue** parameter are displayed.

-filtervalue?
Display a list of valid filter attributes. The valid filters for the **svcinfo lsmdiskgrp** command are:

- name
- storage_pool_id
- mdisk_count
- vdisk_count
- extent_size
- status
- id

Description

This command will return a concise list or a detailed view, of MDisk groups visible to the cluster.

The following list provides possible values that are applicable to the attributes that are displayed as data in the output views:

status online, degraded, offline

The **VDisk_count** field shows the number of VDisk copies that are in the MDisk group.

Command output includes values for the following attributes:

capacity
The total amount of MDisk storage that is assigned to the MDisk group.

free_capacity
The amount of MDisk storage that is assigned to the MDisk group that is unused. MDisk storage can be used for cluster quorum data and VDisks.

real_capacity
The amount of MDisk storage that is assigned to the MDisk group that is assigned to VDisks.

virtual_capacity
The total virtual size of all the VDisk copies that are associated with the MDisk group. This is the same as the real_capacity value unless you have configured space-efficient VDisk copies in this MDisk group.

used_capacity
The total used size of all the VDisk copies that are associated with the MDisk group. This is the same as the real_capacity value unless you have configured space-efficient VDisk copies in this MDisk group.

overallocation
Expressed as a percentage, the ratio of the virtual_capacity value to the capacity. An MDisk group overallocation of over 100 is only possible if you have configured space-efficient VDisk copies.

warning
This field is a percentage. A warning is generated when the amount of space in the MDisk group that has been assigned exceeds this level.

A concise invocation example

```
svcinfo lsmdiskgrp -delim :
```

The concise resulting output

```
id:name:status:mdisk_count:vdisk_count:capacity:extent_size:free_capacity:
virtual_capacity:used_capacity:real_capacity:overallocation:warning
0:mdiskgrp0:degraded:4:0:34.2GB:16:34.2GB:0:0:0:0:0
1:mdiskgrp1:online:4:6:200GB:16:100GB:400GB:75GB:100GB:200:80
```

A detailed invocation example

```
svcinfo lsmdiskgrp -delim : 0
```

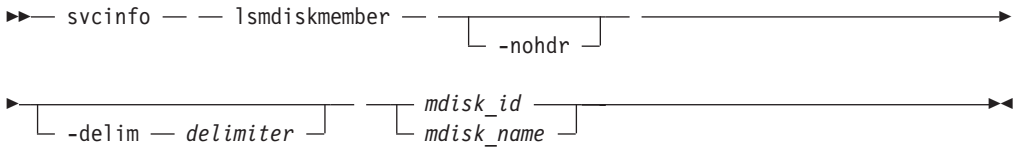
The detailed resulting output

```
id:0
name:mdiskgrp0
status:online
mdisk_count:2
vdisk_count:26
capacity:33.3GB
extent_size:16
free_capacity:26.8GB
virtual_capacity:6.00GB
used_capacity:10.56GB
real_capacity:6.00GB
overallocation:18
warning:0
```

lsmdiskmember

The **lsmdiskmember** command displays a list of VDisks that are using extents on the specified MDisk. That is, the virtual disks are using extents on the managed disk that are specified by the MDisk ID.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise

view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

mdisk_id | mdisk_name
(Required) Specifies the ID or name of the MDisk for which you want a list of VDisks that use extents of that MDisk.

Description

This command displays a list of virtual disks that use extents on the managed disk that are specified by the ID. The list displays members of the respective object and is independent of the state of the individual members; that is, if they are in offline state, they are still displayed.

Every VDisk is constructed from one or more MDisks. To determine the relationship between VDisk copies and MDisks, issue the following command:

```
svcinfo lsvdiskmember vdisk_id | vdisk_name
```

where *vdisk_id | vdisk_name* is the name or ID of the VDisk copy. This displays a list of IDs that correspond to the MDisks that make up the VDisk copy.

To determine the relationship between VDisk copies and MDisks and the number of extents that are provided by each MDisk, you must use the command-line interface. For each VDisk copy, issue the following command:

```
svcinfo lsvdiskextent vdisk_id | vdisk_name
```

where *vdisk_id | vdisk_name* is the name or ID of the VDisk copy. This command displays a table of MDisk IDs and the corresponding number of extents that each MDisk provides as storage for the VDisk copy.

To determine the relationship between MDisks and VDisk copies, issue the following command:

```
svcinfo lsmdiskmember mdisk_id | mdisk_name
```

where *mdisk_id | mdisk_name* is the name or ID of the MDisk. This command displays a list of IDs that correspond to the VDisk copies that are using this MDisk.

To determine the relationship between MDisks and VDisk copies and the number of extents that are used by each VDisk copy, you must use the command-line interface. For each MDisk *mdisk_id | mdisk_name*, issue the following command:

```
svcinfo lsmdiskextent mdisk_id | mdisk_name
```

where *mdisk_id | mdisk_name* is the name or ID of the MDisk. This command displays a table of VDisk copy IDs and the corresponding number of extents that are being used by each VDisk copy.

An invocation example

```
svcinfo lsmdiskmember -delim : 1
```

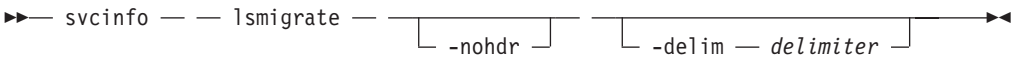
The resulting output

```
id:copy_id
0:0
1:0
2:0
3:0
4:0
5:0
6:0
```

lsmigrate

The **lsmigrate** command displays the progress of all current data migration operations.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

If you use multiple threads to migrate data, the progress will increment when all threads have completed the migration of an extent. For large extent sizes with many threads, this can result in quite large increments in the percentage progress.

Description

This command displays information of all the migrations that are currently in progress.

An invocation example

```
svcinfo lsmigrate -delim :
```

The resulting output

```
migrate_type:MDisk_Group_Migration
progress:96
migrate_source_vdisk_index:33
migrate_target_mdisk_grp:4
max_thread_count:4
migrate_source_vdisk_copy_id:1
```

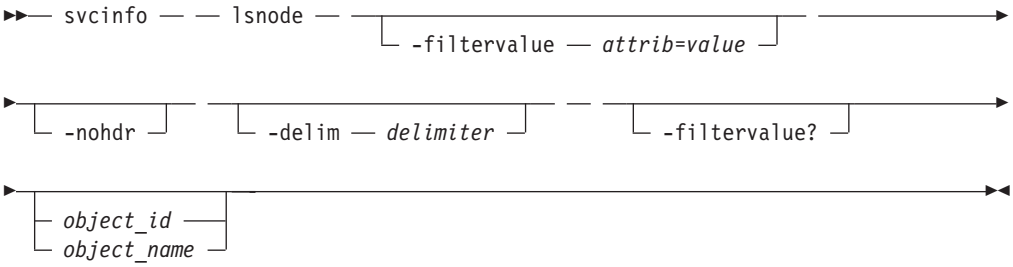
lsnode

The **lsnode** command returns a concise list or a detailed view of nodes visible to the cluster.

The list report style can be used to obtain two styles of report:

- A list containing concise information about all the nodes on a cluster. (Each entry in the list corresponds to a single node.)
- The detailed information about a single node.

Syntax



Parameters

-filtervalue *attrib=value*

(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are returned. If a capacity is specified, the units must also be included.

Note: Some filters allow the use of a wildcard when you enter the command. The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard.
- When using a wildcard, you must enclose the filter entry within double quotation marks (""), as follows:

```
svcinfo lsnode -filtervalue "name=md*"
```

-filtervalue?

Displays a list of valid filter attributes for the **-filtervalue** *attribute=value* parameter. The valid filters for the **svcinfo lsnode** command are:

- *node_name*
- *id*
- *status*
- *IO_group_name*
- *IO_group_id*
- *name*
- *hardware*

-nohdr

(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | object_name
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id | object_name* parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

Description

This command returns a concise list or a detailed view of nodes visible to the cluster. Table 15 provides the possible values that are applicable to the attributes that are displayed as data in the output views.

Table 15. svcinfo lsnode attribute values. Describes the attribute values for the svcinfo lsnode command

Attribute	Value
status	**offline | flushing | pending | online | adding | deleting**
config_ node	**no | yes**
port_status	**active | inactive | not installed**
hardware	**8A4 | 8G4 | 8F2 | 8F4 | 4F2 | other**

To determine a node's WWPNs, enter the following command:

```
svcinfo lsnode
```

Note: Record the node name or ID because you use this value in the next step. For the node or nodes in question, issue the following command:

```
svcinfo lsnode node_name | node_id
```

where *node_name | node_id* is the node name or ID.

Note: Record the four port IDs (WWPNs).

Attention: If the node is in the adding state, the WWPN is displayed as 0000000000000000. Once the node has successfully become a member of the cluster, the state changes to online and the WWPN is displayed correctly.

A concise invocation example

```
svcinfo lsnode -delim :
```

The concise resulting output

```
id:name: UPS_serial_number: WWNN:
 status: IO_group_id: IO_group_name:
 config_node: UPS_unique_id: hardware
1:node1: UPS_SN: 5005076801000024:
        online: 0: io_grp0: yes:
 1000000000000024: other
```

A detailed invocation example

```
svcinfo lsnode -delim= 1
```

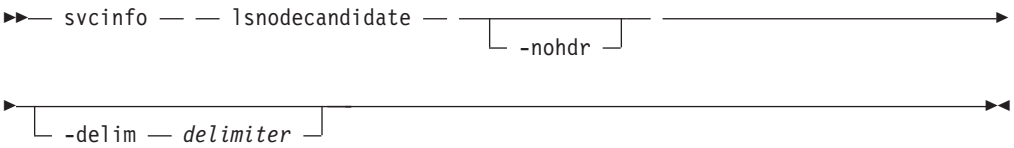
The detailed resulting output

```
id=1
name=node1
UPS_serial_number=UPS_Fake_SN
WWNN=50050768010007E5
status=online
IO_group_id=0
IO_group_name=io_grp0
partner_node_id=
partner_node_name=
config_node=yes
UPS_unique_id=10000000000007E5
port_id=50050768011007E5
port_status=active
port_speed=2
port_id=50050768012007E5
port_status=inactive
port_speed=2
port_id=50050768013007E5
port_status=not_installed
port_speed=2
port_id=50050768014007E5
port_status=not_installed
port_speed=2
hardware=8F2
```

lsnodecandidate

The **lsnodecandidate** command lists all of the nodes that are not assigned to a cluster.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command displays a list of nodes that are not assigned to a cluster.

An invocation example

```
svcinfo lsnodecandidate -delim :
```

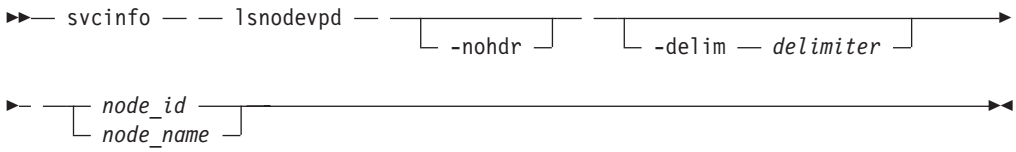
The resulting output

```
id: panel_name: UPS_serial_number:
 UPS_unique_id: hardware
500507680100000EF: oneg55:
 UPS_Fake_SN:100000000000000EF: other
```

lsnodevpd

The **lsnodevpd** command displays the vital product data (VPD) for each node.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

node_id | node_name
(Required) Specifies the node by its ID or name.

Description

This command displays the VPD for the specified node. Each field is reported on a new line. All fields are strings. The VPD is split into sections. Each section has a section heading. The number of fields in that section follows each heading. Each section is separated by an empty line.

For example:

section name:3 fields
field1:value
field2:value
field3:value

new section:x fields
...

Some sections contain information about multiple objects of that type. Each object within the section is separated by an empty line.

For example:

section name:4 fields
object1 field1:value
object1 field2:value

object2 field1:value
object2 field2:value

new section: x fields
...

Note: For 8F4, 8G4, and 8A4 nodes, the VPD displays the device serial number of the fibre-channel card as `N/A`.

An invocation example

```
svcinfo lsnodevpd 1
```

The resulting output

```
id 1

system board: 17 fields

part_number Unknown
system_serial_number 550117N
number_of_processors 2
number_of_memory_slots 4
number_of_fans 0
number_of_FC_cards 1

number_of_scsi/ide_devices 2
BIOS_manufacturer IBM
BIOS_version -[QAE115AUS-1.01]-
BIOS_release_date 08/16/2001
system_manufacturer IBM
```

```
system_product eserver xSeries 342 -[86691RX]-
planar_manufacturer IBM
power_supply_part_number Unknown
CMOS_battery_part_number Unknown
power_cable_assembly_part_number Unknown
service_processor_firmware N/A

processor: 6 fields
processor_location Processor 1
number_of_caches 2

manufacturer GenuineIntel
version Pentium III
speed 1000
status Enabled
processor cache: 4 fields
type_of_cache Internal L1 Cache
size_of_cache (KB) 32

type_of_cache Internal L2 Cache
size_of_cache (KB) 256

processor: 6 fields
processor_location Processor 2
number_of_caches 2
manufacturer GenuineIntel
version Pentium III
speed 1000
status Enabled

processor cache: 4 fields
type_of_cache Internal L1 Cache
size_of_cache (KB) 32

type_of_cache Internal L2 Cache
size_of_cache (KB) 256
memory module: 16 fields
part_number 33L5039
device_location J1
bank_location Slot1 in bank 1
size (MB) 1024
part_number 33L5039
device_location J4
bank_location Slot2 in bank 1
size (MB) 1024

part_number N/A
device_location J2
bank_location Slot1 in bank 2
size (MB) 0

part_number N/A
device_location J3
bank_location Slot2 in bank 2
size (MB) 0

FC card: 5 fields
part_number 64P7783
port_numbers 1 2
device_serial_number VSI 0000AD3F4
manufacturer Agilent
device DX2
device: 15 fields
part_number Unknown
bus ide0
device 0
model LG CD-ROM CRN-8245B
```

```
revision 1.13
serial_number
approx_capacity 0
part_number Unknown
bus scsi
device 0
device_vendor IBM-ESXS
model ST318305LC    !#
revision 6C48
serial_number 3JKQ93B903196C48
approx_capacity 8
software: 5 fields
code_level 00000000
node_name node1
ethernet_status 1
WWNN 0x50050768010007e5

id 1

front panel assembly: 3 fields
part_number Unknown
front_panel_id lynn02
front_panel_locale en_US

UPS: 10 fields
electronics_assembly_part_number FakElec
battery_part_number FakBatt
frame_assembly_part_number FakFram
input_power_cable_part_number FakCabl
UPS_serial_number UPS_Fake_SN
UPS_type Fake UPS
UPS_internal_part_number UPS_Fake_PN
UPS_unique_id 0x10000000000007e5
UPS_main_firmware 1.4
UPS_comms_firmware 0.0
```

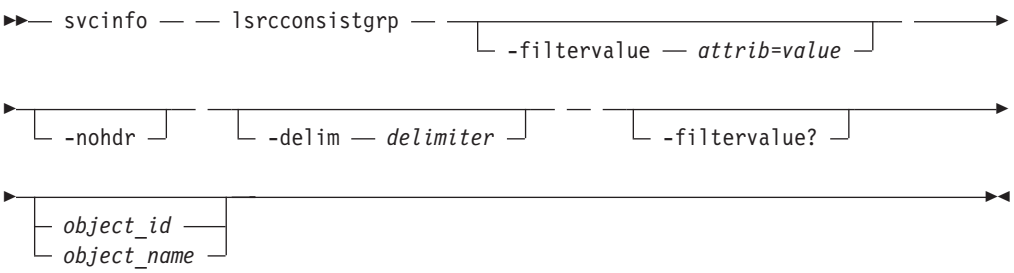
lsrcconsistgrp

The **lsrcconsistgrp** command returns a concise list or a detailed view of Metro or Global Mirror consistency groups visible to the cluster.

The list report style can be used to obtain two styles of report:

- A list containing concise information about all the Metro or Global Mirror consistency groups visible to the cluster. (Each entry in the list corresponds to a single Metro or Global Mirror consistency group.)
- The detailed information about a single Metro or Global Mirror consistency group.

Syntax



Parameters

-filtervalue *attrib=value*

(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are displayed. If a capacity is specified, the units must also be included.

Note: Some filters allow the use of a wildcard when you enter the command. The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard, which must be the first or last character in the string.
- When using a wildcard, you must enclose the filter entry with double quotation marks (""), as follows:

```
svcinfo lsrcconsistgrp -filtervalue "name=md*"
```

-nohdr

(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*

(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | object_name

(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is displayed and any value that is specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id | object_name* parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

-filtervalue?

(Optional) Specifies that you want your report to display any or all of the list of valid filter attributes. The following filter attributes for the **svcinfo lsrcconsistgrp** command are valid:

- group_id
- name
- master_cluster_id
- master_cluster_name
- aux_cluster_id
- aux_cluster_name
- primary
- state

- relationship_count
- id
- copy_type

Description

This command returns a concise list or a detailed view of Metro or Global Mirror consistency groups that are visible to the cluster.

Table 16 provides possible values for the attributes that are displayed as data in the output views.

Table 16. **lsrcconsistgrp** *command output values.* Describes **lsrcconsistgrp** output data attribute values

Attribute	Value
primary	n/a, master, aux
state	inconsistent_stopped, inconsistent_copying, consistent_stopped, consistent_synchronized, idling, idling_disconnected, inconsistent_disconnected consistent_disconnected, empty
freeze_time	The time in YY/MM/DD/HH/MM format.
status	online, primary_offline, secondary_offline
sync	in_sync, out_of_sync
copy_type	metro, global, empty_group

Note: The names of the Metro or Global Mirror relationships and consistency groups might be blank if the relationship or consistency groups are intercluster and the cluster partnership is disconnected.

A concise invocation example

```
svcinfo lsrcconsistgrp -delim :
```

The concise resulting output

```
id:name:master_cluster_id:master_cluster_name:aux_cluster_id:aux_cluster_name:pr
 imary:state:relationship_count:copy_type

 248:jdemo_BA_cons1:0000020060406746:clusterB:0000020061413ABA:clusterA:master:co
 nsistent_stopped:2:global
 249:rccstgrp0:0000020061413ABA:clusterA:0000020061413ABA:clusterA::empty:0
:empty_group
 250:jdemo_BA_cons2:0000020060406746:clusterB:0000020061413ABA:clusterA:master:in
 consistent_stopped:1:metro
 251:BA_cons1:0000020060406746:clusterB:0000020061413ABA:clusterA:master:consiste
 nt_stopped:4:metro
 252:AB_cons2:0000020061413ABA:clusterA:0000020060406746:clusterB::empty:0
:empty_group
 253:AB_cons1:0000020061413ABA:clusterA:0000020060406746:clusterB:aux:consistent_
 stopped:3:global
 254:AA_cons2:0000020061413ABA:clusterA:0000020061413ABA:clusterA::empty:0
:empty_group
 255:AA_cons1:0000020061413ABA:clusterA:0000020061413ABA:clusterA:master:consiste
 nt_synchronized:2:global
```

A detailed invocation example

```
svcinfo lsrcconsistgrp -delim : 254
```

The detailed resulting output

```
id:254
name:rccstgrp0
master_cluster_id:0000010030A007E5
master_cluster_name:kkk
aux_cluster_id:0000010030A007E5
aux_cluster_name:kkk
primary:master
state:inconsistent_stopped
relationship_count:1
freeze_time:
status:online
sync:
copy_type:metro
RC_rel_id:2
RC_rel_name:aaa
```

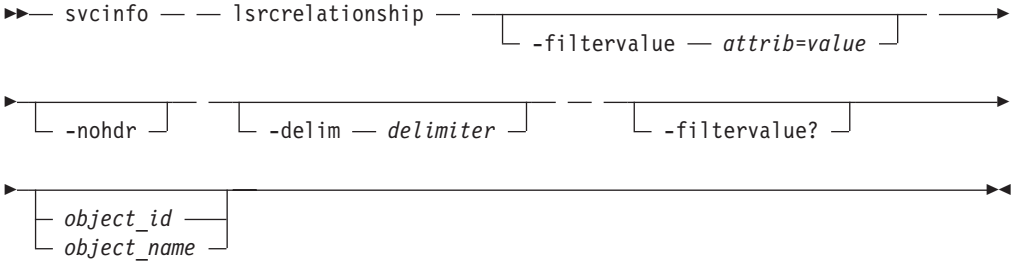
lsrcrelationship

The **lsrcrelationship** command returns a concise list or a detailed view of Metro or Global Mirror relationships visible to the cluster.

The list report style can be used to obtain two styles of report:

- A list containing concise information about all the Metro or Global Mirror relationships visible to the cluster. (Each entry in the list corresponds to a single Metro or Global Mirror relationship.)
- The detailed information about a single Metro or Global Mirror relationship.

Syntax



Parameters

-filtervalue *attribute=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are returned. If a capacity is specified, the units must also be included.

Note: Some filters allow the use of a wildcard when you enter the command. The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard, which must be the first or last character in the string.
- When using a wildcard, you must enclose the filter entry with double quotation marks (" "), as follows:

```
svcinfo lsrcrelationship -filtervalue  "name=md*"
```

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed even if the **-nohdr** parameter is specified.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | object_name
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id | object_name* parameter, the concise view of all objects matching the filtering requirements that are specified by the **-filtervalue** parameter are displayed.

-filtervalue?
(Optional) Specifies that you want your report to display any or all of the list of valid filter attributes. The valid filter attributes for the **svcinfo lsrcrelationship** command are:

- RC_rel_id
- RC_rel_name
- master_cluster_id
- master_cluster_name
- master_vdisk_id
- master_vdisk_name
- aux_cluster_id
- aux_cluster_name
- aux_vdisk_id
- aux_vdisk_name
- primary
- consistency_group_id
- consistency_group_name
- state
- progress
- copy_type

Description

This command returns a concise list or a detailed view of Metro or Global Mirror relationships visible to the cluster.

Table 17 provides possible values for the attributes that are displayed as data in the output views.

Table 17. **lsrcrelationship** *command attributes and values.* Describes the **lsrcrelationship** output data

Attribute	Value
primary	n/a, master, aux
state	inconsistent_stopped, inconsistent_copying, consistent_stopped, consistent_synchronized, idling, idling_disconnected, inconsistent_disconnected, consistent_disconnected
progress	0-100, n/a
freeze time	The time in YY/MM/DD/HH/MM format.
status	online, primary_offline, secondary_offline
sync	n/a, in_sync, out_of_sync
copy_type	metro, global

Note: The names of the Metro or Global Mirror relationships and consistency groups can be blank if the relationship or consistency groups are inter-cluster and the cluster partnership is disconnected.

A concise and detailed invocation example

```
svcinfo lsrcrelationship -delim : -filtervalue name=j*
```

The concise and detailed resulting output

```
id:name:master_cluster_id:master_cluster_name:master_vdisk_id:master_vdisk_name:
aux_cluster_id:aux_cluster_name:aux_vdisk_id:
aux_vdisk_name:primary:consistency_group_id:consistency_group_name:state:bg_copy
_priority:progress: copy_type
45:jrel_AB1:0000020061413ABA:clusterA:45:jdisk_B8:0000020060406746:clusterB:38:j
disk_B1:master:::consistent_stopped:50:metro
48:jrel_AB2:0000020061413ABA:clusterA:48:jdisk_A4:0000020060406746:clusterB:41:j
disk_B4:master:::consistent_synchronised:50:metro
49:jrel_BA_1:0000020060406746:clusterB:42:jdisk_B5:0000020061413ABA:clusterA:49:j
disk_A5:master:248:jdemo_BA_cons1:consistent_stopped:50:metro
50:jrel_BA_2:0000020060406746:clusterB:43:jdisk_B6:0000020061413ABA:clusterA:
50:jdisk_A6:master:248:jdemo_BA_cons1:consistent_stopped:50:metro
```

A detailed invocation example

```
svcinfo lsrcrelationship -delim : AB_2
```

The detailed resulting output

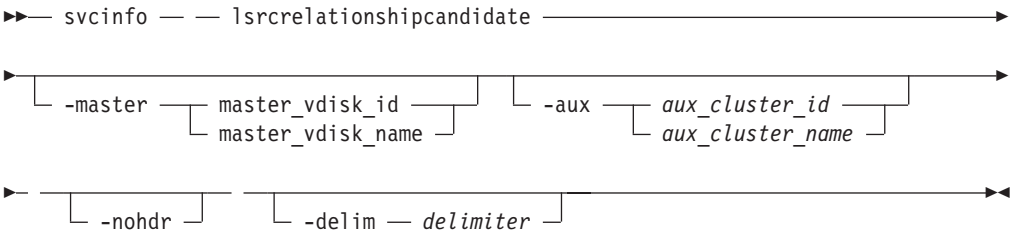
```
id:9
 name:AB_2
 master_cluster_id:0000020061413ABA
 master_cluster_name:clusterA
 master_vdisk_id:9
 master_vdisk_name:stripe9
 aux_cluster_id:0000020060406746
 aux_cluster_name:clusterB
 aux_vdisk_id:9
 aux_vdisk_name:stripe9_b
 primary:master
 consistency_group_id:
 consistency_group_name:
 state:consistent_stopped
 bg_copy_priority:50
 progress:
```

```
 freeze_time:2006/05/05/08/26/46
 status:secondary_offline
 sync:in_sync
copy_type:metro
```

lsrcrelationshipcandidate

The **lsrcrelationshipcandidate** command lists VDisks that are eligible to form Metro or Global Mirror relationships. You can list eligible VDisks that are on the local or remote cluster.

Syntax



Parameters

-master *master_vdisk_id* | *master_vdisk_name*
(Required) Specifies a particular VDisk to use as the master VDisk. The command finds candidates that match the size of this VDisk. If you are requesting candidate VDisks on the local cluster, this command also matches the io_group.

-aux *aux_cluster_id* | *aux_cluster_name*
(Required) Specifies a remote cluster with VDisk candidates for an intercluster relationship. If you do not specify this parameter, the candidates on the local cluster are displayed.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command displays a list of VDisks that can be either the master or the auxiliary disk for a Metro or Global Mirror relationship. VDisk IDs and names are displayed.

An invocation example

```
svcinfo lsrcrelationshipcandidate -delim :
```

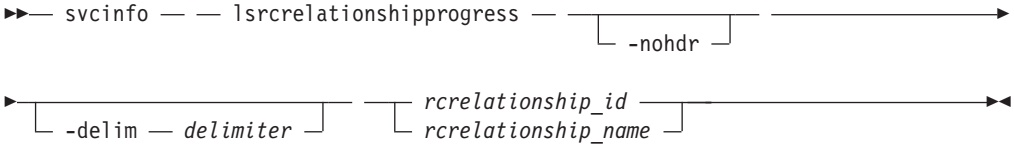
The resulting output

```
id:vdisk_name
0:vdisk0
4:vdisk4
```

lsrcrelationshipprogress

You can use the **lsrcrelationshipprogress** command to display the progress of the background copy of a Metro Mirror or Global Mirror relationship as a percentage. When the initial background copy process for a relationship has completed, null is displayed for the progress of that relationship.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

rcrelationship_id | *rcrelationship_name*
Specifies the object ID or name of the specified type.

Description

This command displays the progress of the background copy of a Metro Mirror or Global Mirror relationship as a percentage.

An invocation example

```
svcinfo lsrcrelationshipprogress -delim : 0
```

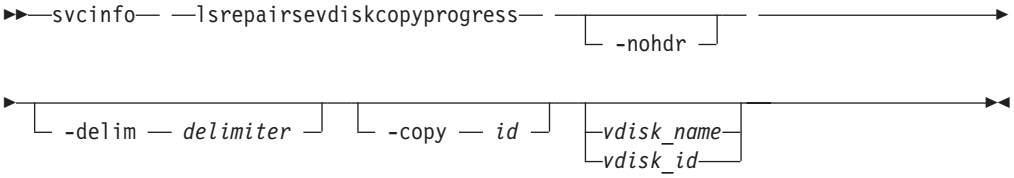
The resulting output

```
id:progress
0:58
```

lsrepairsevdiskcopyprogress

The **lsrepairsevdiskcopyprogress** command lists the repair progress for space-efficient VDisk copies.

Syntax



Parameters

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-copy *id*
(Optional) Lists the repair progress for the specified copy.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

vdisk_name | *vdisk_id*
(Optional) Specifies the virtual disk name or ID to list repair progress for. You must specify this parameter last on the command line. If you do not enter this parameter, the command lists progress for all space-efficient copies in the cluster.

Description

The **lsrepairsevdiskcopyprogress** command lists the repair progress for space-efficient copies of the specified VDisk. If you do not specify a VDisk, the command lists the repair progress for all space-efficient copies in the cluster.

Note: Only run this command after running the **svctask repairsevdiskcopy** command, which you must only run as required by the Directed Maintenance Procedures or by IBM support.

An invocation example

```
svcinfo lsrepairsevdiskcopyprogress –delim :
```

The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:repairing:50:070301120000
0:vdisk0:1:repairing:51:070301120000
1:vdisk1:0:repairing:32:070301153500
```

An invocation example

```
svcinfo lsrepairsevdiskcopyprogress –delim : vdisk0
```

The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:repairing:50:070301120000
0:vdisk0:1:repairing:51:070301120000
```

An invocation example

```
svcinfo lsrepairsevdiskcopyprogress –delim : -copy 1 vdisk0
```

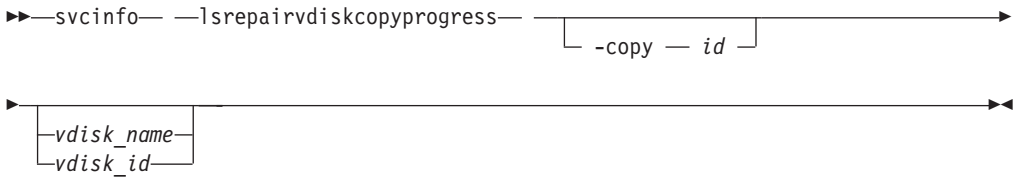
The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:1:repairing:51:070301120000
```

lsrepairvdiskcopyprogress

The **lsrepairvdiskcopyprogress** command displays the progress of mirrored VDisk repairs and validations.

Syntax



Parameters

-copy *id*
(Optional) Specifies the VDisk copy ID to list repair progress for. If you do not specify this parameter, progress is displayed for all copies.

vdisk_name | *vdisk_id*
(Optional) Specifies the virtual disk name or ID to list repair progress for. You must specify this parameter last on the command line.

Description

The **lsrepairvdiskcopyprogress** command displays the progress of repairs and validations being made to mirrored VDisks. Use this command to track progress after running the **svctask repairvdiskcopy** command. You can specify a VDisk copy using the **-copy** *id* parameter. To display the VDisks that have two or more copies with an active task, specify the command with no parameters; it is not possible to have only one VDisk copy with an active task.

The command displays progress for the following types of VDisk copies:

- All VDisk copies display the same task; validate, medium or resync, depending on the specified parameter.
- All VDisk copies display the same percentage and estimated completion time.
- If specified, non-mirrored VDisks are displayed as a single copy with a blank task; they are not displayed in the full concise view.
- Once a task completes, the task is blank for all copies.
- If the task is blank, the percentage and the completion time are also blank.

An invocation example

```
svcinfo lsrepairvdiskcopyprogress –delim :
```

The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:medium:50:070301120000
0:vdisk0:1:medium:50:070301120000
```

An invocation example

```
svcinfo lsrepairvdiskcopyprogress –delim : vdisk0
```

The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:medium:50:070301120000
0:vdisk0:1:medium:50:070301120000
```

An invocation example

```
svcinfo lsvdiskcopyrepairprogress –delim : -copy 0 vdisk0
```

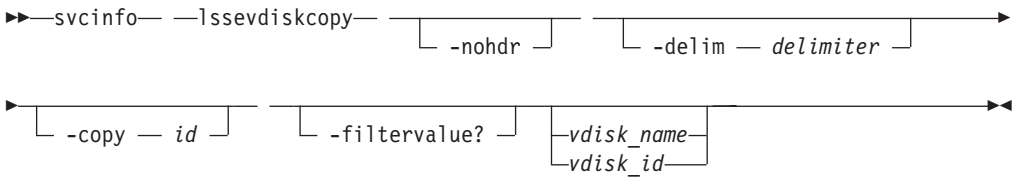
The resulting output

```
vdisk_id:vdisk_name:copy id:task:progress:estimated_completion_time
0:vdisk0:0:medium:50:070301120000
```

lssevdiskcopy

The **lssevdiskcopy** command lists the space-efficient copies of the specified VDisks.

Syntax



Parameters

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the

command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-copy *id*
(Optional) Specifies the VDisk copy to list space-efficient copies for. You must specify a *vdisk_name | vdisk_id* value with this parameter.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-filtervalue?
(Optional) Displays a list of valid filter attributes. The following filters for the **svcinfo lssevdiskcopy** command are valid:

- mdisk_grp_id
- mdisk_grp_name
- overallocation
- autoexpand
- grainsize

vdisk_name | vdisk_id
(Optional) Specifies the virtual disk name or ID to list space-efficient copies for. You must specify this parameter last on the command line. If you do not enter this parameter, the command lists all space-efficient VDisk copies in the cluster.

Description

The **lssevdiskcopy** command lists all space-efficient copies of the specified VDisk. If you do not specify a VDisk, the command lists all space-efficient copies in the cluster.

The command provides a concise view of the space-efficient properties of the selected VDisk copies. Run the **svcinfo lsvdiskcopy** command to see a concise view of the properties that are common to space-efficient and non-space-efficient VDisk copies. See the description of **svcinfo lsvdisk** command for a description of the fields that is shown in the view.

An invocation example

```
svcinfo lssevdiskcopy –delim :
```

The resulting output

```
vdisk_id:vdisk_name:copy_id:mdisk_grp_id:mdisk_grp_name:capacity:used_capacity:real
_capacity:
free_capacity:overallocation:autoexpand:warning:grainsize
0:vv1:0:0:ppp:16.0GB:5.0MB:4.0GB:15.99GB:400:off:20:32
1:se1:0:0:ppp:16.0GB:1.0GB:4.0GB:15.00GB:400:off:20:32
1:se1:1:0:ppp:16.0GB:2.0GB:8.0GB:14.00GB:200:off:45:256
```

An invocation example

```
svcinfo lssevdiskcopy –delim : se1
```

The resulting output

```
vdisk_id:vdisk_name:copy_id:mdisk_grp_id:mdisk_grp_name:capacity:used_capacity:real
_capacity:
free_capacity:overallocation:autoexpand:warning:grainsize
1:se1:0:0:ppp:16.0GB:1.0GB:4.0GB:15.00GB:400:off:20:32
1:se1:1:0:ppp:16.0GB:2.0GB:8.0GB:14.00GB:200:off:45:256
```

An invocation example

```
svcinfo lssevdiskcopy -delim : -copy 0 0
```

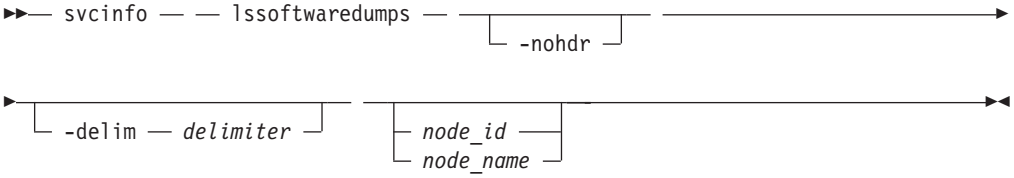
The resulting output

```
id:0
name:vdisk0
capacity:128.0MB
copy_id:0
status:online
sync:yes
primary:yes
mdisk_grp_id:0
mdisk_grp_name:mdiskgrp0
type:striped
mdisk_id:
mdisk_name:
fast_write_state:empty
used_capacity:0.41MB
real_capacity:128.00MB
free_capacity:127.59MB
overallocation:100
autoexpand:off
warning:79
grainsize:32
```

lssoftwaredumps

Use the **lssoftwaredumps** command to display a list of software packages from the **/home/admin/upgrade** directory.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the

-delim parameter is a one-byte character. If you enter **-delim :** on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

node_id | node_name
(Optional) Specifies the node ID or name to list the available dumps of the specific type. If you do not specify a node, the dumps that are available on the configuration node are listed.

Description

This command displays a list of software upgrade packages. These packages are copied as a result of software upgrades. If you do not specify a node, the packages that are available on the configuration node are listed. The command displays files from the /home/admin/upgrade directory.

An invocation example

```
svcinfo lssoftwaredumps
```

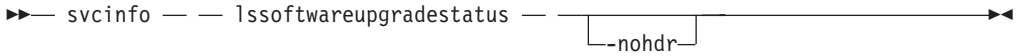
The resulting output

```
id               software_filename
0                s1_mala75_030405_092143
1                s2_mala75_030405_092145
2                s3_mala75_030405_092146
```

lssoftwareupgradestatus

The **lssoftwareupgradestatus** command displays the status of a software upgrade.

Syntax



Parameters

-nohdr
(Optional) Suppresses the display of headings.

Description

The **lssoftwareupgradestatus** command displays the status of a software upgrade.

An invocation example

```
svcinfo lssoftwareupgradestatus
```

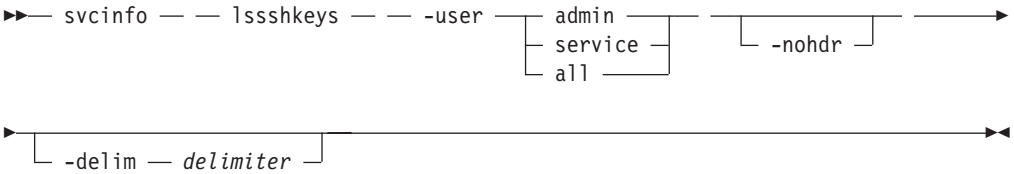
The resulting output

```
status
active
```

lssshkeys

Use the **lssshkeys** command to display a list of SSH (secure shell) keys that are available on the cluster.

Syntax



Parameters

-user admin | service | all
(Required) Specifies whether you want to see a list of keys that can be used by only a service user or an admin user or the list for both user types.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

Description

This command displays a list of all the keys that are available on the cluster for the specified user ID.

An invocation example

```
svcinfo lssshkeys -user all -delim :
```

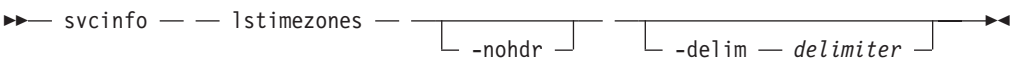
The resulting output

```
id:userid_key:identifier
1:admin:admin
```

lstimezones

The **lstimezones** command lists the time zones that are available on the cluster. Each timezone is assigned an ID that can be used in the **svctask settimezone** command to set the time zone.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

Description

This command displays a list of all the time zones that are available on the cluster. Each time zone is assigned an ID. This ID can be used in the **svctask settimezone** command.

An invocation example

```
svcinfo lstimezones
```

The resulting output

```
id timezone
0 Africa/Abidjan
1 Africa/Accra
2 Africa/Addis_Ababa
3 Africa/Algiers
4 Africa/Asmera
5 Africa/Bamako
6 Africa/Bangui
```

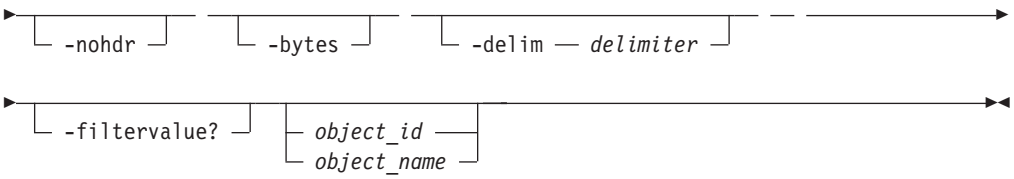
lsvdisk

The **lsvdisk** command displays a concise list or a detailed view of VDisks that are recognized by the cluster.

The list report style can be used to obtain two different styles of report.

- A list containing concise information about all the virtual disks that are recognized by the cluster. (Each entry in the list corresponds to a single virtual disk.)
- The detailed information about a single virtual disk.

Syntax



Parameters

-filtervalue *attrib=value*
(Optional) Specifies a list of one or more filters. Only objects with a value that matches the filter attribute value are displayed. If a capacity is specified, the units must also be included.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-bytes
(Optional) Displays all capacities as bytes.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

object_id | object_name
(Optional) Specifies the name or ID of an object. When you use this parameter, the detailed view of the specific object is returned and any value that is specified by the **-filtervalue** parameter is ignored. If you do not specify the *object_id | object_name* parameter, the concise view of all objects matching the filtering requirements that is specified by the **-filtervalue** parameter are displayed.

-filtervalue?
(Optional) Displays a list of valid filter attributes. The following filters for the **svcinfo lsvdisk** command are valid:

- vdisk_name
- vdisk_id
- vdisk_UID
- fc_map_count
- copy_count
- IO_group_id
- IO_group_name
- status
- mdisk_grp_name
- mdisk_grp_id

- capacity
- type
- FC_id
- FC_name
- RC_id
- RC_name
- name
- id

Note: It is not possible to filter the **svcinfo lsvdisk** command with mdisk_grp_name=many to identify mirrored VDisks. Instead, filter on copy_count=2.

Description

This command displays a concise list or a detailed view of attributes for all VDisks and VDisk copies in the cluster.

The VDisk is offline and unavailable if one of the following take place:

- If both nodes in the I/O group are missing
- None of the nodes in the I/O group that are present can access the VDisk.
- All synchronized copies for this VDisk are in MDisk groups that are offline.

If you have a degraded VDisk and all of the associated nodes and MDisks are online, call the IBM Support Center for assistance. A VDisk is reported as degraded if any of the following occurs:

- One of the nodes in the I/O group is missing.
- One of the nodes in the I/O group cannot access all the MDisks in the MDisk group that the VDisk spans. In this case MDisks are shown as degraded and the DMPs for MDisks should be followed to resolve the problem.
- The fast write cache pins data for one or more VDisks in the I/O group and is unable to perform a failback until the situation is resolved. An error log indicating that the cache has pinned data is displayed. Follow the directed maintenance procedures for this error log to resolve the problem. The most common causes of pinned data are the following:
  - One or more VDisks in an I/O group is offline due to an asymmetric failure and has pinned data in the cache. Asymmetric failures can occur because of SAN fabric faults or misconfiguration, back-end controller faults or misconfiguration or because repeated errors has lead to the cluster excluding access to a MDisk through one or more nodes.
  - One or more VDisks in an I/O group is offline due to a problem with a FlashCopy mapping.

The command returns values for the following VDisk attributes:

IO_groups_id/name
Specifies the I/O Group that the VDisk belongs to.

status The value can be **online**, **offline** or **degraded**.

mdisk_grp_id/name
Specifies the name and ID of the MDisk group that the VDisk belongs to. If the VDisk has more than one copy, these fields display **many**.

type Specifies the virtualization type of the VDisk. The value can be **striped**,

sequential, **image** or **many**. The value **many** indicates that the VDisk has more than one copy, which can have different virtualization types.

capacity
Specifies the total capacity of the VDisk.

formatted
Indicates whether the VDisk was formatted when it was created. The value can be **Yes** or **No**.

mdisk_id/name
Specifies the MDisk that is used for sequential and image mode VDisks. If the VDisk has more than one copy, these fields display **many**.

FC_id/name
Specifies the name and ID of the FlashCopy mapping that the VDisk belongs to. The value **many** indicates that the VDisk belongs to more than one FlashCopy mapping.

RC_id/name
Specifies the name and ID of the Global Mirror or Metro Mirror relationship that the VDisk belongs to.

vdisk_UID
Specifies the UID of the VDisk.

throttling
Specifies the throttle rate of the VDisk.

preferred_node_id
Specifies the ID of the preferred node for the VDisk.

fast_write_state
Specifies the cache state for the VDisk. The value can be **empty**, **not_empty**, **corrupt**, or **repairing**. A cache state of **corrupt** indicates that the VDisk requires recovery by using one of the **recovervdisk** commands. A cache state of **repairing** indicates that repairs initiated by a **recovervdisk** command are in progress.

cache Specifies the cache mode of the VDisk. The value can be **readwrite** or **none**.

udid Specifies the unit number for the VDisk. Only OpenVMS hosts require a unit number.

fc_map_count
Specifies the number of FlashCopy mappings that the VDisk belongs to.

sync_rate
Specifies the rate for synchronization for mirrored copies.

The command returns values for the following VDisk copy attributes:

copy_id
Specifies a system-assigned identifier for the VDisk copy. The value can be **0** or **1**.

status The value can be **online** or **offline**. A copy is offline if all nodes cannot access the MDisk group that contains the copy.

sync Indicates whether the VDisk copy is synchronized.

primary
Indicates whether the VDisk copy is the primary copy. A VDisk has exactly one primary copy. The value can be **Yes** or **No**.

mdiskgrp_id/name
Specifies the name and ID of the MDisk group that the VDisk copy belongs to.

type Specifies the virtualization type of the VDisk. The value can be **striped**, **sequential** or **image**.

mdisk_id/name
Specifies the MDisk that is used for sequential and image mode VDisks.

fast_write_state
Specifies the cache state of the VDisk copy. The value can be **empty**, **not_empty**, **corrupt**, or **repairing**. The value is always empty for non-space-efficient copies. A cache state of **corrupt** indicates that the VDisk is space-efficient and requires repair that is initiated by a **recovervdisk** command or the **repairsevdiskcopy** command.

used_capacity
Specifies the portion of **real_capacity** that is being used to store data. For non-space-efficient copies, this value is the same as the VDisk capacity. If the VDisk copy is space-efficient, the value increases from zero to the **real_capacity** value as more of the VDisk is written to.

real_capacity
Specifies the amount of physical storage that is allocated from an MDisk group to this VDisk copy. If the VDisk copy is not space-efficient, the value is the same as the VDisk capacity. If the VDisk copy is space-efficient, the value can be different.

free_capacity
Specifies the difference between the **real_capacity** and **used_capacity** values.

overallocation
Expressed as a percentage, specifies the ratio of VDisk capacity to **real_capacity** values. This value is always **100** for non-space-efficient VDisks.

autoexpand
Specifies whether **autoexpand** is enabled on a space-efficient VDisk. The value can be **on** or **off**.

warning
Expressed as a percentage, for space-efficient VDisk copies only. A warning is generated when the ratio of **used_capacity** to VDisk capacity reaches the specified level.

grainsize
For space-efficient VDisk copies, specifies the grain size chosen for the VDisk copy when it was created.

A concise invocation example

```
svcinfo lsvdisk -delim :
```

The concise resulting output

```
| id:name:IO_group_id:IO_group_name:status:mdisk_grp_id:mdisk_grp_name:capacity:type:FC_id:
|  FC_name:RC_id:RC_name:vdisk_UID:fc_map_count:copy_count:fast_write_state
| 0:vdisk0:0:io_grp0:degraded:0:mdiskgrp0:16.0MB:striped:::::60050768017F06BF7800000000000000:1
| 1:vdisk1:0:io_grp0:degraded:0:mdiskgrp0:16.0MB:striped:::::60050768017F06BF7800000000000001:1
| 2:vdisk2:0:io_grp0:degraded:0:mdiskgrp0:16.0MB:striped:::::60050768017F06BF7800000000000002:1
| 3:vdisk3:0:io_grp0:degraded:0:mdiskgrp0:16.0MB:striped:::::60050768017F06BF7800000000000003:1
```

A detailed invocation example

```
svcinfo lsvdisk -delim : 251
```

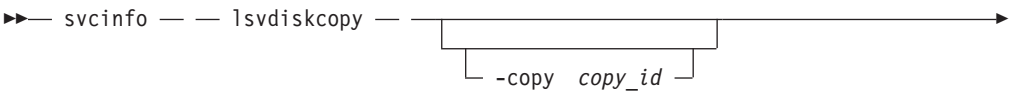
The detailed resulting output

```
id:251
name:i0vd163
IO_group_id:0
IO_group_name:io_grp0
status:online
mdisk_grp_id:3
mdisk_grp_name:vind1
capacity:16.0MB
type:striped
formatted:no
mdisk_id:
mdisk_name:
FC_id:
FC_name:
RC_id:
RC_name:
vdisk_UID:6005076801A0002C800000000000078B
throttling:0
preferred_node_id:1
fast_write_state:empty
cache:readwrite
udid:
fc_map_count:0
sync_rate:50
copy_count:1

copy_id:0
status:online
sync:yes
primary:yes
mdisk_grp_id:3
mdisk_grp_name:vind1
type:striped
mdisk_id:
mdisk_name:
fast_write_state:empty
used_capacity:16.00MB
real_capacity:16.00MB
free_capacity:0.00MB
overallocation:100
autoexpand:
warning:
grainsize:
```

lsvdiskcopy

The **lsvdiskcopy** command lists VDisk copy information.

Syntax



Parameters

-copy *copy_id*
(Optional) Specifies the VDisk copy to list information for. You must specify a *vdisk_name* | *vdisk_id* value with this parameter.

-filtervalue?
(Optional) Displays a list of valid filter attributes. The following filters for the **svcinfo lsvdiskcopy** command are valid:

- primary
- status
- sync
- mdisk_grp_id
- mdisk_grp_name
- type

vdisk_name | *vdisk_id*
(Optional) Specifies the VDisk to list copy information for. You must specify this parameter last on the command line. If you specify a *vdisk_name* | *vdisk_id* value only, all copies for the VDisk are listed.

Description

The **lsvdiskcopy** command lists information for VDisk copies. If you specify the command with no parameters, all VDisks and copies in the cluster are listed.

An invocation example

```
svcinfo lsvdiskcopy -delim :
```

The resulting output

```
vdisk_id:vdisk_name:copy_id:status:sync:primary:mdisk_grp_id:mdisk_grp_name:
capacity:type
0:vd1:0:online:yes:yes:1:mdisk_group_1:20GB:striped
0:vd1:1:offline:no:no:2:mdisk_group_2:20GB:striped
1:vd2:0:online:yes:yes:mdisk_group_2:100GB:image
```

An invocation example

```
svcinfo lsvdiskcopy -delim : vd1
```

The resulting output

```
vdisk_id:vdisk_name:copy_id:status:sync:primary:mdisk_grp_id:mdisk_grp_name:
capacity:type
0:vd1:0:online:yes:yes:1:mdisk_group_1:20GB:striped
0:vd1:1:offline:no:no:2:mdisk_group_2:20GB:striped
```

An invocation example

```
svcinfo lsvdiskcopy -copy 0 –delim : vv1
```

The resulting output

```
id:0
name:vdisk0
capacity:128.0MB
copy_id:0
```

```
status:online
sync:yes
primary:yes
mdisk_grp_id:0
mdisk_grp_name:mdiskgrp0
type:striped
mdisk_id:
mdisk_name:
fast_write_state:empty
used_capacity:0.41MB
real_capacity:128.00MB
free_capacity:127.59MB
overallocation:100
autoexpand:off
warning:79grainsize:32
```

lsvdiskdependentmaps

The **lsvdiskdependentmaps** command displays all FlashCopy mappings with target virtual disks (VDisks) that are dependent upon data held on the specified VDisk.

Syntax

Parameters

vdisk_id | *vdisk_name*
(Required) Specifies the name or ID of a virtual disk (VDisk).

Description

The **lsvdiskdependentmaps** command displays FlashCopy mappings that have target VDisks that are dependent upon data held on the specified *vdisk_id* | *vdisk_name*. This can be used to determine whether a FlashCopy mapping can be prepared. Issue the command for the target VDisk *vdisk_id* | *vdisk_name* of the FlashCopy mapping to be prepared. If no FlashCopy mappings are returned, the FlashCopy mapping can be prepared. Any FlashCopy mappings that are returned in the list must be stopped or be in the **idle_or_copied** state, before the new FlashCopy mapping can be prepared.

A concise invocation example

```
svcinfo lsvdiskdependentmaps -delim : 0
```

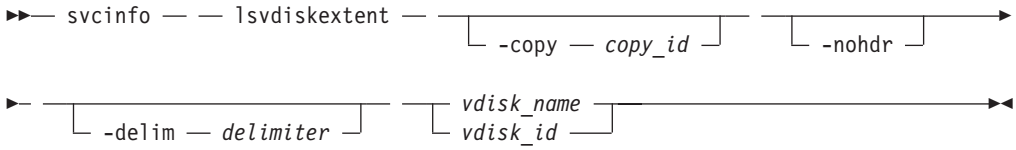
The concise resulting output

```
id:name
2:fcmap2
5:fcmap5
```

lsvdiskextent

The **lsvdiskextent** command lists the MDisk extents that are provided for the specified VDisks.

Syntax



Parameters

-copy *copy_id*
(Optional) Displays a list of MDisks that are members of the specified VDisk copy.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

vdisk_name | *vdisk_id*
(Required) Specifies one or more virtual disk IDs or names.

Description

The **lsvdiskextent** command displays a list of MDisk IDs and the number of extents that each MDisk provides to the specified VDisks.

Each VDisk is constructed from one or more MDisks. To determine the relationship between a VDisk and its MDisks, issue the following command:

```
svcinfo lsvdiskmember vdisk_name | vdisk_id
```

where *vdisk_name* | *vdisk_id* is the name or ID of the VDisk. This command displays a list of MDisk IDs that make up the VDisk.

To determine the number of extents that are provided by each MDisk, issue the following command:

```
svcinfo lsvdiskextent vdisk_name | vdisk_id
```

where *vdisk_name* | *vdisk_id* is the name or ID of the VDisk. This command displays a table of MDisk IDs and the corresponding number of extents that each MDisk provides as storage for the given VDisk.

To determine the relationship between MDisks and VDisks, issue the following command for each MDisk:

```
svcinfo lsmdiskmember mdisk_name | mdisk_id
```

where *mdisk_name* | *mdisk_id* is the name or ID of the MDisk. This command displays a list of IDs that corresponds to the VDisks that are using this MDisk.

To determine the relationship between MDisks and VDisks, and the number of extents that are used by each VDisk, you must use the command-line interface. For each MDisk, issue the following command:

```
svcinfo lsmdiskextent mdisk_name | mdisk_id
```

where *mdisk_name* | *mdisk_id* is the name or ID of the MDisk. This command displays a table of VDisk IDs and the corresponding number of extents that are used by each VDisk.

An invocation example

```
svcinfo lsvdiskextent -delim : vdisk0
```

The resulting output

```
id:number_extents
0:0
```

lsvdiskfcmappings

The **lsvdiskfcmappings** command displays a list of FlashCopy mappings to which the VDisk belongs. A VDisk can be part of up to 256 FlashCopy mappings.

Syntax

Parameters

vdisk_name | *vdisk_id*
(Required) Specifies the name or ID of the VDisk for which a list of all FlashCopy mappings is required.

Description

The **lsvdiskfcmappings** command returns a list of all FlashCopy mappings that the VDisk is a member of. The list is returned in no particular order.

An invocation example

```
svcinfo lsvdiskfcmappings -delim : vdisk2
```

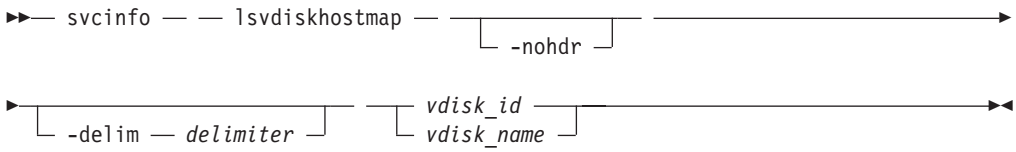
The resulting output

```
fc_id:fc_name
1:fcmap1
3:fcmap3
```

lsvdiskhostmap

Use the **lsvdiskhostmap** command to list the VDisk to the host mapping. These hosts have the specified virtual disk mapped to them; the virtual disk is visible to these hosts.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

vdisk_id | *vdisk_name*
(Required) Specifies the ID or name of the virtual disk. The cluster displays a list of all the hosts to which this virtual disk is mapped and the SCSI ID by which the virtual disk is mapped.

Description

This command displays a list of host IDs and names. These hosts have the specified virtual disk mapped to them; that is, the virtual disk is visible to these hosts. The SCSI LUN ID is also displayed. The SCSI LUN ID is the ID by which the virtual disk is recognized by the host.

Determining the host that a VDisk is mapped to: List the hosts that this VDisk is mapped to, by issuing the following command:

```
svcinfo lsvdiskhostmap vdisk_id | vdisk_name
```

where *vdisk_id* | *vdisk_name* is the name or ID of the VDisk. A list is displayed. Look for the host name or ID to determine which host this VDisk is mapped to. If no data is displayed, the VDisk is not mapped to any hosts.

The concise invocation example

```
svcinfo lsvdiskhostmap -delim : 3
```

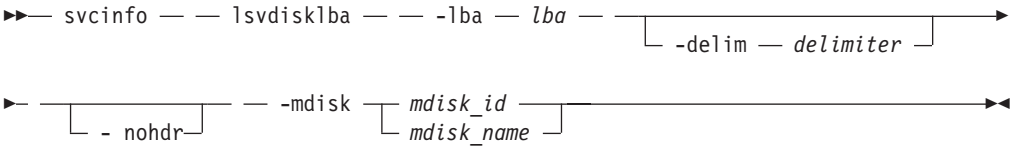
The concise resulting output

```
id:name:SCSI_id:host_id:host_name:wwpn:vdisk_UID
3:vdisk3:0:2:host2:0000000000100ABC:60050768018A0001500000000000003
```

lsvdisklba

The **lsvdisklba** command lists the VDisk and logical block address (LBA) for the specified MDisk LBA.

Syntax



Parameters

-lba *lba*
(Required) Specifies the 64-bit hexadecimal LBA on the MDisk.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

-mdisk *mdisk_id* | *mdisk_name*
(Required) Specifies the MDisk name or ID.

Description

The **lsvdisklba** command returns one line of output that describes the LBA of the VDisk that is associated with the MDisk LBA.

If applicable, the command also lists the range of LBAs on both the VDisk and MDisk that are mapped in the same extent, or for space-efficient disks, in the same grain.

Table 5 on page 105 provides command output that depends on several variables.

Table 18. lsvdisklba command output scenarios. Describes command output scenarios.

Field	Typical scenario	Quorum disk	Space-efficient metadata	Extent not allocated	Formatting extent	Extent allocated to space-efficient disk, LBA not used on space-efficient disk
copy_id	yes	no	yes	no	yes	yes
vdisk_id	yes	no	yes	no	yes	yes
vdisk_name	yes	no	yes	no	yes	yes
type	allocated	metadata	metadata	unallocated	formatting	unallocated
lba	yes	no	no	no	no	no
vdisk_start	yes	no	no	no	no	no
vdisk_end	yes	no	no	no	no	no
mdisk_start	yes	yes	yes	yes	yes	yes
mdisk_end	yes	yes	yes	yes	yes	yes

An invocation example

```
svcinfo lsvdisklba -mdisk 1 -lba 0x3480000 delim :
```

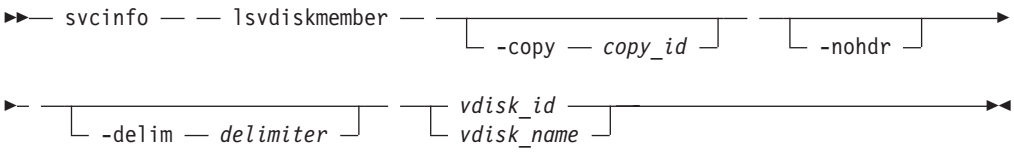
The resulting output

```
vdisk_id:vdisk_name:copy_id:type:lba:vdisk_start:vdisk_end:mdisk_start:mdisk_end
17:vdisk17:0:metadata:::0x3480000:0x34808ff
```

lsvdiskmember

The **lsvdiskmember** command displays a list of MDisks that are members of the specified VDisk.

Syntax



Parameters

-copy *copy_id*
(Optional) Displays a list of MDisks that are members of the specified VDisk copy.

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible

width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

vdisk_id | vdisk_name
(Required) Displays a list of MDisks that are members of the specified VDisk.

Description

This command displays a list of managed disks, which provide extents that make up the virtual disk that is specified by the ID.

Every VDisk is constructed from one or more MDisks. At times, you might have to determine the relationship between the two objects. The following procedure allows you to determine the relationships.

If you use the **svcinfo lsmdiskmember** command, the concise view displays a list of virtual disks. These are the virtual disks that are using extents on the managed disk that is specified by the ID. The list displays the members of the respective object and is independent of the state of the individual members; that is, if they are in offline state, they are still displayed.

To determine the relationship between VDisks and MDisks, issue the following command:

```
svcinfo lsvdiskmember vdisk_id | vdisk_name
```

where *vdisk_id | vdisk_name* is the name or ID of the VDisk. This displays a list of IDs that correspond to the MDisks that make up the VDisk.

To determine he relationship between VDisks and MDisks, and the number of extents that are provided by each MDisk, you must use the command-line interface. Issue the following command:

```
svcinfo lsvdiskextent vdisk_id | vdisk_name
```

where *vdisk_id | vdisk_name* is the name or ID of the VDisk. This displays a table of MDisk IDs and the corresponding number of extents that each MDisk provides as storage for the specified VDisk.

To determine the relationship between MDisks and VDisks, issue the following command:

```
svcinfo lsmdiskmember mdisk_id | mdisk_name
```

where *mdisk_id | mdisk_name* is the name or ID of the MDisk. This displays a list of IDs that correspond to the VDisks that are using this MDisk.

To determine he relationship between MDisks and VDisks, and the number of extents that are used by each VDisk, you must use the command-line interface. For a specified MDisk, issue the following command:

```
svcinfo lsmdiskextent mdisk_id | mdisk_name
```

where *mdisk_id | mdisk_name* is the name or ID of the MDisk. This displays a table of VDisk IDs and the corresponding number of extents that are used by each VDisk.

An invocation example

```
svcinfo lsvdiskmember 1
```

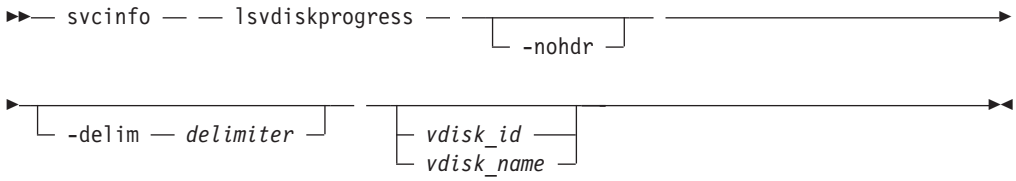
The resulting output

```
id
2
```

lsvdiskprogress

The **lsvdiskprogress** command tracks the progress during new virtual disk formatting.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

vdisk_id | vdisk_name
(Optional) Specifies the VDisk ID or name. If you do not specify this parameter, the progress of all VDisks currently being formatted is displayed.

Description

This command displays the progress of the format of a new virtual disk as a completed percentage. If the VDisk has multiple copies, the command reports the average progress of the format.

An invocation example

```
svcinfo lsvdiskprogress -delim : 0
```

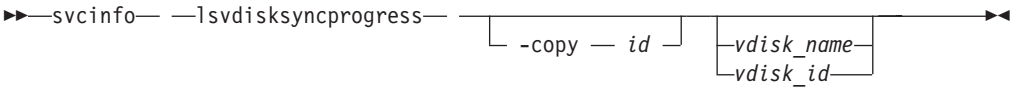
The resulting output

```
id:progress
0:58
```

lsvdisksyncprogress

The **lsvdisksyncprogress** command displays the progress of VDisk copy synchronization.

Syntax



Parameters

-copy *id*
(Optional) Specifies the VDisk copy ID to list synchronization progress for. You must also specify a *vdisk_name | vdisk_id* value. If you do not specify this parameter, progress is displayed for all copies.

vdisk_name | vdisk_id
(Optional) Specifies the virtual disk name or ID to list synchronization progress for.

Description

The **lsvdisksyncprogress** command displays the progress of VDisk copy synchronization. To display the VDisk copies that require synchronization, specify the command with no parameters. The command displays progress for the following types of VDisks:

- A synchronized copy displays a progress of 100 and a blank estimated completion time.
- An offline copy displays a blank estimated completion time, and a gradually decreasing progress if the VDisk is being written to.
- Nonmirrored VDisks are displayed as a single copy with a progress of 100, and a blank estimated completion time.

The **lsvdisksyncprogress** command also displays the progress of a mirrored VDisk synchronization. After you create a mirrored VDisk using the **svctask mkvdisk** or **svctask addvdiskcopy** command, you can use the command to monitor the progress of the synchronization.

An invocation example

```
svcinfo lsvdisksyncprogress
```

The resulting output

```
vdisk_id vdisk_name copy_id progress estimated_completion_time
0 vdisk0 0 100
0 vdisk0 1 50 070301150000
3 vdisk3 0 72 070301132225
```

```
3 vdisk3 1 100
4 vdisk4 0 22 070301160000
4 vdisk4 1 100
8 vdisk8 0 100
8 vdisk8 1 33
```

An invocation example

```
svcinfo lsvdisksyncprogress vdisk0
```

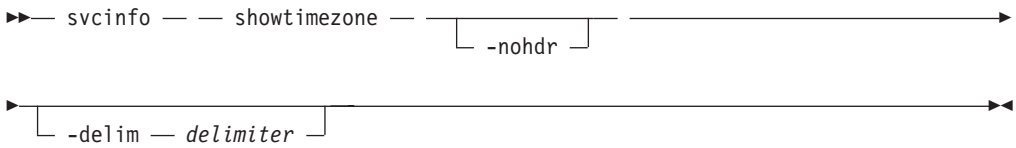
The resulting output

```
vdisk_id vdisk_name copy_id progress estimated_completion_time
0 vdisk0 0 100
0 vdisk0 1 50 070301150000
```

showtimezone

Use the **showtimezone** command to display the current time zone settings for the cluster.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed, the data is separated from the header by a space. The **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified character.

Description

This command displays a single time zone and its associated ID. This is the current time zone setting for the cluster. A list of available time-zones can be viewed by running the **svcinfo lstimezones** command. The time zone can be changed by running the **svctask settimezone** command.

An invocation example

```
svcinfo showtimezone -delim :
```

The resulting output

```
id:timezone
522:UTC
```

Chapter 21. Error log commands

Error log commands capture and manage error conditions for the cluster.

finderr

The **finderr** command analyzes the error log for the unfixed error with the highest severity.

Syntax

Description

The command scans the error log for any unfixed errors. The highest priority unfixed error is returned.

You can use this command to determine the order in which to fix the logged errors.

The Web-based directed maintenance procedures (DMPs) also use this command.

An invocation example

```
svcservicetask finderr
```

The resulting output

```
Highest priority unfixed error code is [1010]
```

dumperrlog

The **dumperrlog** command dumps the contents of the error log to a text file.

Syntax

Parameters

-prefix *filename_prefix*
A file name is created from the prefix and a time stamp, and has the following format:

prefix_NNNNNN_YYMMDD_HHMMSS

where *NNNNNN* is the node front panel name.

Note: If the **-prefix** parameter is not supplied, the dump is directed to a file with a system-defined prefix of **errlog**.

Description

When run with no parameters, the **dumperrlog** command dumps the cluster error log to a file using a system-supplied prefix of **errlog**, which includes the node ID and time stamp. When a file name prefix is provided, the same operation is performed but the details are stored in the dumps directory within a file with a name that starts with the specified prefix.

A maximum of ten error-log dump files are kept on the cluster. When the eleventh dump is made, the oldest existing dump file is overwritten.

Error log dump files are written to **/dumps/elogs**. The contents of this directory can be viewed using the **svcinfo lserrlogdumps** command.

Files **are not** deleted from other nodes until you issue the **cleardumps** command.

An invocation example

```
svcservicetask dumperrlog -prefix testerrorlog
```

The resulting output

```
No feedback
```

clearerrlog

The **clearerrlog** command clears all entries from the error log including status events and any unfixed errors.

Syntax

Parameters

-force
(Optional) Specifies that the function of the **clearerrlog** command be processed without confirmation requests. If the **-force** parameter is not supplied, you are prompted to confirm that you want to clear the log.

Description

This command clears all entries from the error log. The entries are cleared even if there are unfixed errors in the log. It also clears any status events that are in the log.

Attention: This command is destructive. Use it only when you have either rebuilt the cluster or have fixed a major problem that has caused entries in the error log that you do not want to manually fix.

An invocation example

```
svcservicetask clearerrlog -force
```

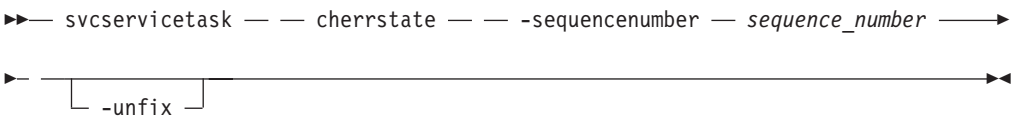
The resulting output

```
No feedback
```

cherrstate

The **cherrstate** command marks an unfixed error as fixed. You can also use it to mark a fixed error as unfixed.

Syntax



Parameters

-sequencenumber *sequence_number*
(Required) Specifies the error log sequence numbers to fix or mark as unfixed.

-unfix
(Optional) Specifies that the sequence numbers that are supplied be marked as unfixed. If you specify the **-unfix** parameter, the sequence numbers are marked as unfixed. Use this parameter when you have marked the wrong sequence number as fixed.

Description

This command allows you to specify error log entries as fixed or unfixed. Use this command as a manual confirmation step that you have performed a maintenance procedure on the cluster, fabric, or subsystems.

This step is performed as part of the directed maintenance procedures (DMPs).

Optionally, if you have marked the wrong sequence number as fixed, you can use the **-unfix** parameter to change the entry to unfixed.

An invocation example

```
svcservicetask cherrstate -sequencenumber 2019
```

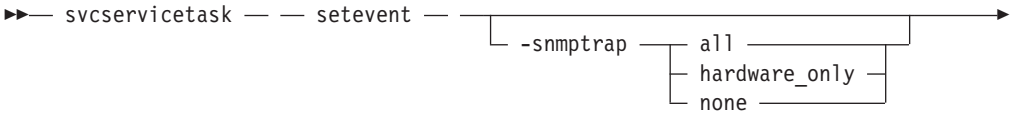
The resulting output

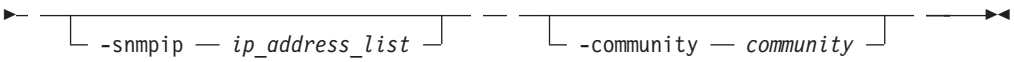
```
No feedback
```

setevent

The **setevent** command specifies how the SAN Volume Controller sends Simple Network Management Protocol (SNMP) traps when an error or event is logged to the error log.

Syntax





Parameters

-snmptrap all | hardware_only | none
(Optional) Specifies the SNMP trap setting, which specifies when to receive a message that reports a problem or significant event. You can set the following values for this parameter:

all Sends an SNMP trap for all errors and state changes that are logged.

hardware_only
Sends an SNMP trap for all errors, but not for object state changes.

none Does not send any SNMP traps or errors. This is the default setting for a new cluster.

-snmpip *ip_address_list*
(Optional) Specifies the IP addresses of host systems running the SNMP manager software. You can specify up to six IP addresses, using one of following formats:

- Colon-separated list of IPv4 addresses
- Comma-separated list of IPv6 addresses
- Comma-separated list of IPv4 or IPv6 addresses, including an optional port number for each address. For example:
  - For IPv4: 9.20.83.124:80,9.20.83.125:81
  - For IPv6: [FFE0:1234::1]:80,[FFE0:1234::2]:81

Entries in excess of the number specified using the **-community** parameter are ignored.

-community *community*
(Optional) Specifies the SNMP community string. This is a colon-separated list of values with up to six items per list. The maximum length of the community string that is used in SNMP trap generation cannot be more than 60 characters.

Description

This command sets or modifies error log settings, which define the actions to take when errors and events are logged.

You can use this command to set up SNMP traps. For SNMP, you must supply the following information:

- When to raise a trap
- The IP address of the SNMP manager
- The SNMP community

Table 19 provides IPv4 and IPv6 formats that are supported for the **-snmpip** *ip_address_list* parameter:

Table 19. SNMP ip_address_list formats

IP type	*ip_address_list* **format**
IPv4 (no port set, SAN Volume Controller uses default port of 162)	`1.2.3.4`
IPv4 with specific port	`1.2.3.4:22`

Table 19. SNMP ip_address_list formats (continued)

IP type	*ip_address_list* **format**
Full IPv6 (no port set, SAN Volume Controller uses default port of 162)	`2002:914:fc12:848:209:6bff:fe8c:4ff6`
Full IPv6 with port	`[2002:914:fc12:848:209:6bff:fe8c:4ff6]:23`
Zero-compressed IPv6 (no port set, SAN Volume Controller uses default port of 162)	`2002::4ff6`
Zero-compressed IPv6 with port	`[2002::4ff6]:23`

An invocation example

```
svcservicetask setevent -snmptrap all
```

The resulting output

```
No feedback
```

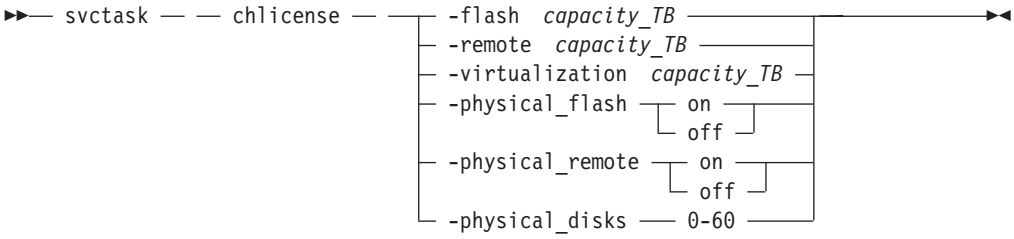
Chapter 22. Licensing commands

The following commands enable you to work with SAN Volume Controller licensed functions.

chlicense

The **chlicense** command changes license settings for cluster features.

Syntax



Parameters

-flash *capacity_TB*
(Optional) Changes cluster licensing for the FlashCopy feature. To change the licensed capacity for the FlashCopy feature, specify a capacity in terabytes (TB).

-remote *capacity_TB*
(Optional) Changes cluster licensing for the Metro Mirror and Global Mirror feature. To change the licensed capacity for the Metro Mirror and Global Mirror feature, specify a capacity in terabytes (TB).

-virtualization *capacity_TB*
(Optional) Changes cluster licensing for the Virtualization feature. To change the licensed capacity for the Virtualization feature, specify a capacity in terabytes (TB).

-physical_flash on | off
(Optional) For physical disk licensing, enables or disables the FlashCopy feature. The default value is **off**.

-physical_remote on | off
(Optional) For physical disk licensing, enables or disables the Metro Mirror and Global Mirror feature. The default value is **off**.

-physical_disks 0-60
(Optional) Changes the licensed settings of the cluster for physical disk licensing. Enter the number of physical disks in your cluster. Valid values are 0 - 60.

Note:

- If the **-physical_disks** value is set to zero, the **-physical_flash** and **-physical_remote** values are turned off.
- If the **-physical_disks** value is nonzero, the **-flash**, **-remote**, and **-virtualization** values cannot be set.

- If the **-flash**, **-remote**, or **-virtualization** values are nonzero, the **-physical_flash**, **-physical_remote**, and **-physical_disks** values cannot be set.
- If the **-physical_disks** value is nonzero, only the FlashCopy and RemoteCopy usage is monitored and appropriate error messages are logged.

Description

The **chlicense** command changes license settings for the cluster. Any change that is made is logged as an event in the license setting log.

SAN Volume Controller version 4.3.1 provides two license options: physical disk licensing and capacity licensing. To select physical disk licensing, run the **svctask chlicense** command with the **-physical_disks** parameter. To select capacity licensing, run the **svctask chlicense** command with the **-flash**, **-remote**, or **-virtualization** parameter.

The current license settings for the cluster are displayed in the viewing license settings log panel. These settings show which features you are licensed to use. They also show the storage capacity that is licensed for virtualization. Typically, the license settings log contains entries because feature options must be set as part of the Web-based cluster creation process.

Note: Dumping an empty license settings log produces a file that contains headers, 256 lines of formatted zeros, and two lines that include a checksum operation.

By default, the Copy Services functions are disabled, but this does not stop you from creating and using copy services. However, errors are placed in the license settings log that state that you are using an unlicensed feature. The command-line tool return code also notifies you that you are using an unlicensed feature.

The total virtualized capacity can also be modified with this command. This is the number of terabytes (TB) of virtual disk capacity that can be configured by the cluster.

When you reach 90% capacity, any attempt to create or extend Virtual Disks, Relationships, or Mappings results in a message from the command-line tool. This does not stop you from creating and expanding Virtual Disks, Relationships, or Mappings. When usage reaches or exceeds 100% capacity, errors are placed in the license settings log.

Any error that is placed in the license settings log results in a generic error being placed in the cluster error log. This occurs when you issue a command that violates the license agreement. The return code also notifies you that you are violating the license settings.

An invocation example

```
svctask chlicense -flash 5
```

The resulting output

```
No feedback
```

dumpinternallog

The **dumpinternallog** command dumps the contents of the license settings error and event log to a file on the current configuration node.

Syntax

Description

This command dumps the contents of the internal license settings error and event log to a file on the current configuration node.

This file is always called **feature.txt** and is created, or overwritten, in the **/dumps/feature** directory on the configuration node.

This file can be requested by IBM service personnel.

Before making any entries, the license settings log contains only zeros. A dump of this log from the **svctask dumpinternallog** command results in an empty file.

An invocation example

```
svctask dumpinternallog
```

The resulting output

```
No feedback
```

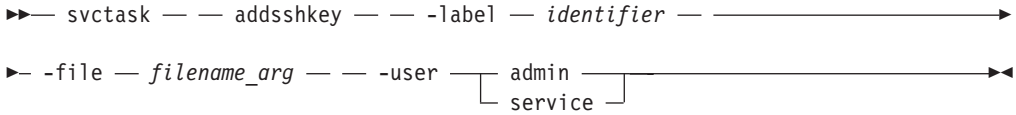
Chapter 23. Secure Shell key commands

Secure Shell (SSH) key commands manage the public and private SSH keys for a cluster.

addsshkey

The **addsshkey** command installs a new SSH key on the cluster. The key file must first be copied onto the cluster.

Syntax



Parameters

-label *identifier*
(Required) Specifies a new ID to associate with this key. The maximum length of this key is 30 characters.

-file *filename_arg*
(Required) Specifies the name of the file containing the SSH key.

-user admin | service
(Required) Specifies which user ID that the SSH key is assigned to.

Description

To add an SSH key to a cluster, you must first copy the key file to the cluster in the **/tmp** directory using secure copy (**scp**). The **svctask addsshkey** command moves the key file from the **/tmp** directory to the required location and activates it for the designated user.

Each key is associated with an ID string that you define, up to 30 characters. Up to 100 keys can be stored on a cluster. You can add keys to provide either administrator access or service access.

When the key has been activated, you can invoke commands on the cluster over SSH using the user ID that was specified from the host for which the key was generated. Alternatively, you can run an interactive SSH session from the specified host using the user ID specified

The identifier (key label) can be used to subsequently identify the key when all keys are listed using the **svcinfo lssshkeys** command, or if the key is to be deleted, using the **svctask rmsshkey** command. The identifier is also specified as the key label that is used in the audit log. The audit log uses the key label to identify the commands that were issued through an SSH session that was established with an associated key. If you want to use the audit log, you must ensure that there is a 1:1 ratio mapping from the key label identifier to the key for each authorized user.

You can issue the **svcinfo lssshkeys** command to list the SSH keys that are available on the cluster. You can issue the **svctask rmsshkey** command to remove an SSH key from the cluster.

You can issue the **svctask rmallsshkeys** command to remove all of the SSH keys from the cluster

An invocation example

```
svctask addsshkey -user service -file /tmp/id_rsa.pub -label testkey
```

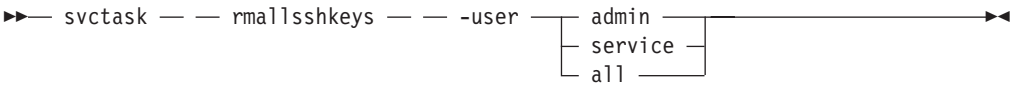
The resulting output

```
No feedback
```

rmallsshkeys

The **rmallsshkeys** command removes all the SSH keys that are specified by the **-user** parameter.

Syntax



Parameters

-user admin | service | all
(Required) Specifies which SSH keys to remove from the cluster. If you specify **-user all**, all the SSH keys in the cluster are removed.

Description

This command removes all the SSH keys that are specified by the **-user** parameter.

You can issue the **svcinfo lssshkeys** command to list the SSH keys that are available on the cluster. You can issue the **svctask addsshkey** command to install a new SSH key on the cluster. The key file must first be copied onto the cluster. Each key is associated with an ID, which can consist of up to 30 characters. Up to 100 keys can be stored on a cluster. You can add keys to provide either administrator access or service access. For example, type the following:

```
svctask addsshkey -user service -file /tmp/id_rsa.pub -label testkey
```

where */tmp/id_rsa.pub* is the name of the file that the SSH key is saved in and *testkey* is the label to associate with this key.

You can issue the **svctask rmsshkey** command to remove an SSH key from the cluster.

An invocation example

```
svctask rmallsshkeys -user service
```

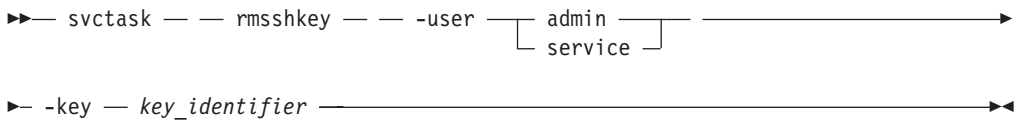
The resulting output

```
No feedback
```

rmsshkey

The **rmsshkey** command removes a specified SSH key from the cluster.

Syntax



Parameters

-user admin | service
(Required) Specifies the user ID from which the SSH key is deleted.

-key *key_identifier*
(Required) Specifies the label of the key to be deleted.

Description

This command removes the specified SSH key from the cluster.

Attention: To change your SSH key, you must add your new key before you remove the old key. Within a current SSH session, once the SSH key that established the session is removed, that session is unable to issue any further commands. Adding your new key before you remove the old key prevents inaccurate audit log entries.

You can issue the **svcinfo lssshkeys** command to list the SSH keys that are available on the cluster.

You can issue the **svctask addsshkey** command to install a new SSH key on the cluster.

You can issue the **svctask rmallsshkeys** command to remove all of the SSH keys from the cluster.

An invocation example

```
svctask rmsshkey  -key testkey -user service
```

The resulting output

```
No feedback
```

Chapter 24. Service mode commands

Service mode commands perform tasks when the node is in service mode. Some of these tasks are to specify node software, to clean dump directories and to dump the contents of an error log to a file.

These commands can only be run on a node that is in service mode. Attempting to run them on a working configuration node results in the message: `CMMVC5997E This command can only be run on a node that is in service mode.`

Attempting to run any of the other **svctask** and **svcservicetask** commands on a node that is in service mode results in the following message: `CMMVC5998E This command can not be run on a node that is in service mode.`

applysoftware

The **applysoftware** command upgrades the cluster to a new level of software.

Syntax

Parameters

-file *filename_arg*
(Required) Specifies the filename of the new software package.

-ignore
(Optional) Specifies that all prerequisite checking be bypassed and that all hardened data be deleted. Hardened data has been written to an auxiliary disk so that it is not lost in the event of a system failure. Use this parameter with caution.

Description

This command starts the upgrade process of the cluster to a new level of software and can be applied to **svcservicetask** and **svcservicemodetask**. The **applysoftware** command can be used to apply a level of software to the node in both service and nonservice modes. In service mode the **applysoftware** command is applied to the specific node in service mode. In nonservice mode, the command is applied to the complete cluster.

The software package as specified by the file name must first be copied on to the current configuration node in the **/home/admin/upgrade** directory. You can use PuTTy secure copy (scp) to copy the file. See "PuTTY scp" for detailed information on this procedure.

The actual upgrade completes asynchronously.

You can view the contents of the **/home/admin/upgrade** directory by using the **svcservicemodeinfo lssoftwaredumps** command.

Internally, the new package is moved from the **/home/admin/upgrade** directory and has a checksum performed on it. If the package fails the checksum operation, it is deleted and the upgrade fails. Otherwise, the package is extracted from the directory and the software upgrade begins.

An invocation example

```
svcservicemodetask applysoftware -file newsoftware
```

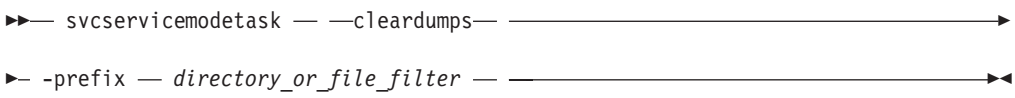
The resulting output

```
No feedback
```

cleardumps

The **cleardumps** command cleans the various dump directories on the node that is in service mode.

Syntax



Parameters

-prefix *directory_or_file_filter*
(Required) Specifies the directory, files, or both to be cleaned. If a directory is specified, with no file filter, all relevant dump or log files in that directory are cleaned. You can use the following directory arguments (filters):

- /dumps (cleans all files in all subdirectories)
- /dumps/configs
- /dumps/elogs
- /dumps/feature
- /dumps/iostats
- /dumps/iotrace
- /home/admin/upgrade

In addition to the directory, you can specify a file filter. For example, if you specify `/dumps/elogs/*.txt`, all files in the **/dumps/elogs** directory that end in `.txt` are cleaned.

Note: The following rules apply to the use of wildcards with the SAN Volume Controller CLI:

- The wildcard character is an asterisk (*).
- The command can contain a maximum of one wildcard.
- When you use a wildcard, enclose the filter entry within double quotation marks (""), as follows:

  ```
  svcservicemodetask cleardumps -prefix "/dumps/elogs/*.txt"
  ```

Description

This command deletes all the files that match *directory_or_file_filter* value for the node that is in service mode.

You can clean all the dumps directories by specifying **/dumps** as the directory value.

You can clean all the files in a single directory by specifying one of the directory values.

You can list the contents of these directories on the given node by using the **svcservicemodeinfo lsxxxxdumps** commands.

You can use this command to clean specific files in a given directory by specifying a directory or file name. You can use the wildcard (*) as part of your file name.

Note: To preserve the configuration and trace files, any files that match the following wildcard patterns are not cleaned:

- *svc.config*
- *.trc
- *.trc.old

An invocation example

```
svcservicemodetask cleardumps -prefix /dumps/configs
```

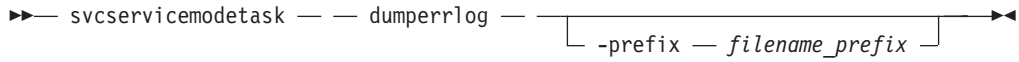
The resulting output

```
No feedback
```

dumperrlog

The **dumperrlog** command dumps the contents of the error log to a text file.

Syntax



Parameters

-prefix *filename_prefix*
(Optional) Creates a file name from the prefix and a time stamp, and has the following format:

prefix_NNNNNN_YYMMDD_HHMMSS

where *NNNNNN* is the node front panel name.

Note: If the **-prefix** parameter is not supplied, the dump is directed to a file with a system-defined prefix of **errlog**.

Description

With no parameters, the **svcservicemodetask dumperrlog** command dumps the cluster error log to a file using a system-supplied prefix of **errlog**, which includes the node ID and time stamp. When a file name prefix is provided, the same operation is performed, but the details are stored in the dumps directory within a file with a name that starts with the specified prefix.

A maximum of ten error log dump files are kept on the cluster. When the 11th dump is made, the oldest existing dump file is overwritten.

Error log dump files are written to the **/dumps/elogs** directory. You can view the contents of this directory using the **svcinfo lserrlogdumps** command.

Files are *not* deleted from other nodes until you issue the **cleardumps** command.

An invocation example

```
svcservicemodetask dumperrlog -prefix testerrorlog
```

The resulting output

```
No feedback
```

exit

The **exit** command exits service mode and restarts the node.

Syntax

Description

This command causes the node to be restarted. The node powers on in standard operating mode and attempts to rejoin the cluster.

At some point during the processing of this command, the SSH (secure shell) client software and the Web server connection that accessed the node are ended as a result of the restart processing.

An invocation example

```
svcservicemodetask exit
```

The resulting output

```
[SSH / webserver connections terminate so an error message to the effect of
'connection lost' may be displayed, or 'CLIENT RECEIVED SERVER DOWN
NOTIFICATION']
```

Chapter 25. Service mode information commands

Service mode information commands perform information gathering tasks when the node is in service mode.

These commands can only be run on a node that is in service mode. Attempting to run them on a working configuration node will result in the message:

CMMVC5997E This command can only be run on a node that is in service mode.

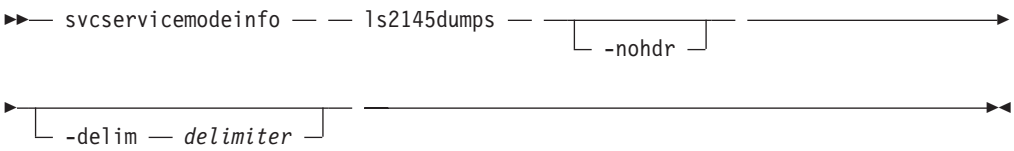
Attempting to run any of the other svcinfo commands on a node that is in service mode will result in the following message:

CMMVC5998E This command can not be run on a node that is in service mode.

ls2145dumps

The **ls2145dumps** command lists node assert dumps and associated output files that are located in the **/dumps** directory.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

Description

This command lists node assert dumps and associated output files, which are created during an assertion of a node. If you do not specify a node, the dumps that are available on the configuration node are listed. The command displays the files that are located in the **/dumps** directory.

An invocation example

```
svcservicemodeinfo ls2145dumps -delim :
```

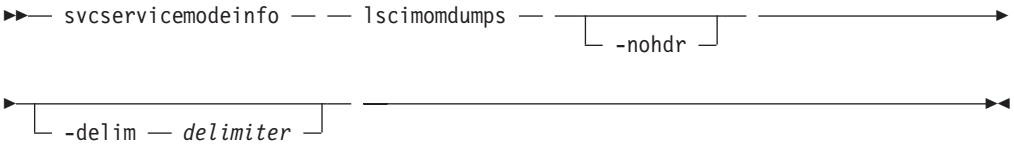
The resulting output

```
id:2145_filename
0:000108.trc.old
1:dump.000108.030328.144007
2.000108.trc
```

lscimomdumps

The **lscimomdumps** command lists the dump files in the **/dumps/cimom** directory.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
By default in a concise view, all columns of data are space-separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row; if the headers are displayed, the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

Description

This command displays a list of Common Information Model object manager (CIMOM) dumps. These dumps are created when you use a CIM client with the CIMOM of the SAN Volume Controller cluster. If you do not specify a node, the available dumps on the configuration node are listed. The command displays the files that are located in the **/dumps/cimom** directory.

An invocation example

```
svcservicemodeinfo lscimomdumps
```

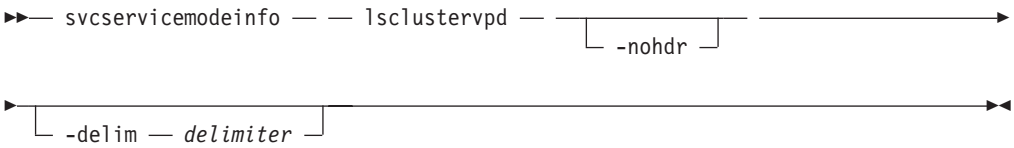
The resulting output

```
id              cimom_filename
0               mkrepositorylog.004565
1               PegasusTrace.004565
2               PegasusStandard.004565
3               PegasusAudit.004565
4               PegasusError.004565
5               PegasusDebug.004565
```

lsclustervpd

The **lsclustervpd** command returns the vital product data (VPD) for the cluster to which the node belonged.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

Description

This command displays the VPD of the cluster to which the node belonged.

An invocation example

```
svcservicemodeinfo lsclustervpd
```

The resulting output

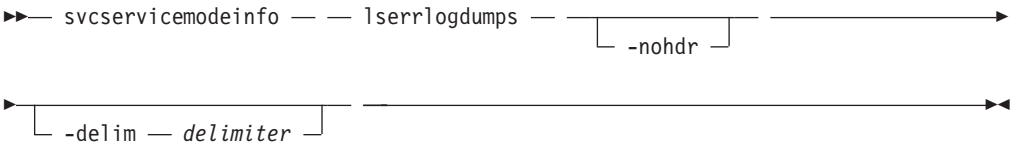
```
id 000001002FF007E5
name kkk
location local
partnership
```

```
bandwidth 0
cluster_IP_address 0.0.0.0
cluster_service_IP_address 1.1.1.1
total_mdisk_capacity 0
space_in_mdisk_grps 0
space_allocated_to_vdisks 0
total_free_space 0
statistics_status off
statistics_frequency 15
required_memory 2048
cluster_locale en_US
SNMP_setting all
SNMP_community
SNMP_server_IP_address 0.0.0.0
subnet_mask 0.0.0.0
default_gateway 0.0.0.0
time_zone 522 UTC
email_setting all
email_id
code_level 00000000
FC_port_speed 1Gb
gm_link_tolerance:300
gm_inter_cluster_delay_simulation:0
gm_intra_cluster_delay_simulation:0
```

lserrlogdumps

The **lserrlogdumps** command lists the error log dump files in the **/dumps/elogs** directory, which are created when you run the **svctask dumperrlog** command.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed even if the **-nohdr** parameter is specified.

-delim *delimiter*
By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

Description

This command lists error log dump files, which are created when you run the **svctask dumperrlog** command. An error log dump describes the contents of the error log at the time that the command was run. If you do not specify a node, the dumps that are available on the configuration node are listed. The command displays the files that are located in the **/dumps/elogs** directory.

An invocation example

```
svcservicemodeinfo lserrlogdumps
```

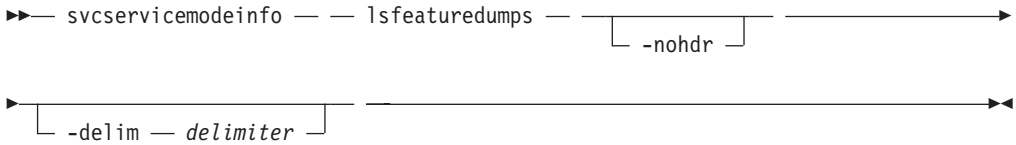
The resulting output

```
id               filename
0                errlog_lynn02_030327_154511
1                aaa.txt_lynn02_030327_154527
2                aaa.txt_lynn02_030327_154559
3                errlog_lynn02_030403_110628
```

lsfeaturedumps

The **lsfeaturedumps** command lists the dump files in the **/dumps/feature** directory, which are created when you run the **svctask dumpinternallog** command.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command returns a list of featurization dumps, which are created when you run the **svctask dumpinternallog** command. A featurization dump file describes the contents of the featurization log at the time that the command was executed. If

you do not specify a node, the available dumps on the configuration node are listed. The command displays files from the **/dumps/feature** directory.

Issue the **svcinfo lsfeaturedumps** command to list the dump files in the **/dumps/feature** directory. The feature log is maintained by the cluster. The feature log records events that are generated when license parameters are entered or when the current license settings have been breached.

An invocation example

```
svcservicemodeinfo lsfeaturedumps
```

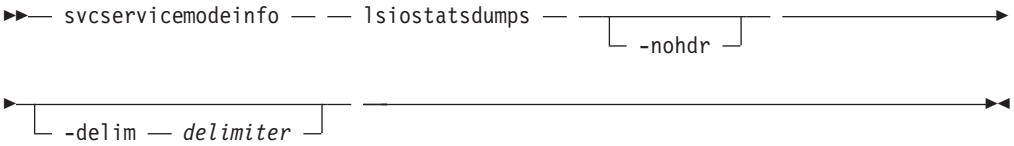
The resulting output

```
id              feature_filename
0               feature.txt
```

lsiostatsdumps

The **lsiostatsdumps** command lists the dump files in the **/dumps/iostats** directory, which are created when you run the **svctask startstats** command.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

Description

This command lists I/O statistics dumps, which are created when you run the **svctask startstats** command. If you do not specify a node, the available dumps on the configuration node are listed. The command displays the files located in the **/dumps/iostats** directory.

An invocation example

```
svcservicemodeinfo lsiostatsdumps
```

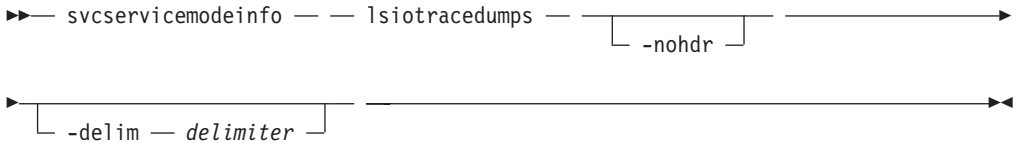
The resulting output

```
id              iostat_filename
0               v_stats_mala75_031123_072426
1               m_stats_mala75_031123_072425
```

lsiotracedumps

The **lsiotracedumps** command lists the files that are located in the **/dumps/iotrace** directory.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by the specified delimiter.

Description

This command lists I/O trace dumps, which are created when you run the **svctask settrace** command. If you do not specify a node, the dump files that are available on the configuration node are listed. The command displays the files that are located in the **/dumps/iotrace** directory.

An invocation example

```
svcservicemodeinfo lsiotracedumps
```

The resulting output

```
id              iotrace_filename
0               c1_mala75_030405_092155
1               c2_mala75_030405_092156
2               c3_mala75_030405_092158
3               c4_mala75_030405_092159
4               c5_mala75_030405_092201
```

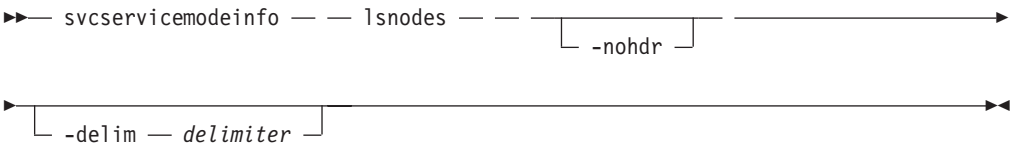
lsnodes

The **lsnodes** command lists the cluster nodes that are in service mode.

The list report style can be used to obtain two styles of report:

- A list containing concise information about all the nodes on a cluster. (Each entry in the list corresponds to a single node.)
- The detailed information about a single node.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one-byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

Description

This command lists the cluster nodes that are in service mode.

A concise invocation example

```
svcservicemodeinfo lsnodes -delim :
```

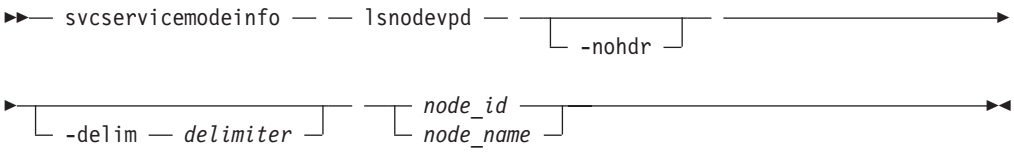
The concise resulting output

```
id:WWNN:front_panel_id:node_name:cluster:fabric
1:50050768010007E5:lynn02:node1:yes:yes
```

lsnodevpd

The **lsnodevpd** command returns the vital product data (VPD) for the specified node.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
(Optional) By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (`:`) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

node_id | *node_name*
(Required) Specifies the node to view in terms of its node ID or name.

Description

This command returns the VPD for the specified node. Each field is reported on a new line. All fields are strings. The VPD is split into sections, each with a section heading. The number of fields in the section follows the heading. Each section is separated by an empty line. For example:

```
section name:3 fields
field1:value
field2:value
field3:value

new section:x fields
...
```

Some sections contain information about multiple objects of that type. Each object within the section is separated by an empty line. For example:

```
section name:4 fields
object1 field1:value
object1 field2:value

object2 field1:value
thanobject2 field2:value
```

new section: x fields
...

Note: For 8F4, 8G4, and 8A4 nodes, the VPD displays the device serial number of the FC card as N/A.

An invocation example

```
svcservicemodeinfo lsnodevpd id 1
```

The resulting output

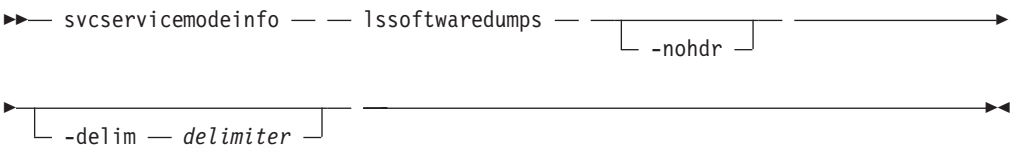
```
id 1

system board: 17 fields

part_number Unknown
system_serial_number 550117N
number_of_processors 2
number_of_memory_slots 4
number_of_fans 0
number_of_FC_cards 1

number_of_scsi/ide_devices 2
BIOS_manufacturer IBM
BIOS_version -[QAE115AUS-1.01]-
BIOS_release_date 08/16/2001
system_manufacturer IBM
system_product eserver xSeries 342 -[86691RX]-
planar_manufacturer IBM
power_supply_part_number Unknown
CMOS_battery_part_number Unknown
power_cable_assembly_part_number Unknown
service_processor_firmware N/A

processor: 6 fields
processor_location Processor 1
number_of_caches 2

manufacturer GenuineIntel
version Pentium III
speed 1000
status Enabled
processor cache: 4 fields
type_of_cache Internal L1 Cache
size_of_cache (KB) 32

type_of_cache Internal L2 Cache
size_of_cache (KB) 256

processor: 6 fields
processor_location Processor 2
number_of_caches 2
manufacturer GenuineIntel
version Pentium III
speed 1000
status Enabled

processor cache: 4 fields
type_of_cache Internal L1 Cache
size_of_cache (KB) 32

type_of_cache Internal L2 Cache
size_of_cache (KB) 256
memory module: 16 fields
```

```
part_number 33L5039
device_location J1
bank_location Slot1 in bank 1
size (MB) 1024
part_number 33L5039
device_location J4
bank_location Slot2 in bank 1
size (MB) 1024

part_number N/A
device_location J2
bank_location Slot1 in bank 2
size (MB) 0

part_number N/A
device_location J3
bank_location Slot2 in bank 2
size (MB) 0

FC card: 5 fields
part_number 64P7783
port_numbers 1 2
device_serial_number VSI 0000AD3F4
manufacturer Agilent
device DX2
device: 15 fields
part_number Unknown
bus ide0
device 0
model LG CD-ROM CRN-8245B
revision 1.13
serial_number
approx_capacity 0
part_number Unknown
bus scsi
device 0
device_vendor IBM-ESXS
model ST318305LC    !#
revision 6C48
serial_number 3JKQ93B903196C48
approx_capacity 8
software: 5 fields
code_level 00000000
node_name node1
ethernet_status 1
WWNN 0x50050768010007e5

id 1

front panel assembly: 3 fields
part_number Unknown
front_panel_id lynn02
front_panel_locale en_US

UPS: 10 fields
electronics_assembly_part_number FakElec
battery_part_number FakBatt
frame_assembly_part_number FakFram
input_power_cable_part_number FakCabl
UPS_serial_number UPS_Fake_SN
UPS_type Fake UPS
UPS_internal_part_number UPS_Fake_PN
UPS_unique_id 0x10000000000007e5
UPS_main_firmware 1.4
UPS_comms_firmware 0.0
```

lssoftwaredumps

The **lssoftwaredumps** command lists the software packages that are located in the **/home/admin/upgrade** directory.

Syntax



Parameters

-nohdr
(Optional) By default, headings are displayed for each column of data in a concise style view, and for each item of data in a detailed style view. The **-nohdr** parameter suppresses the display of these headings.

Note: If there is no data to be displayed, headings are not displayed.

-delim *delimiter*
By default in a concise view, all columns of data are space separated. The width of each column is set to the maximum possible width of each item of data. In a detailed view, each item of data has its own row, and if the headers are displayed the data is separated from the header by a space. Using the **-delim** parameter overrides this behavior. Valid input for the **-delim** parameter is a one byte character. If you enter `-delim :` on the command line, the colon character (**:**) separates all items of data in a concise view; for example, the spacing of columns does not occur. In a detailed view, the data is separated from its header by a colon character.

Description

This command lists software upgrade packages, which are copied as a result of software upgrades. If you do not specify a node, the packages that are available on the configuration node are listed. The command displays files that are located in the **/home/admin/upgrade** directory.

An invocation example

```
svcservicemodeinfo lssoftwaredumps
```

The resulting output

```
id              software_filename
0               s1_mala75_030405_092143
1               s2_mala75_030405_092145
2               s3_mala75_030405_092146
```

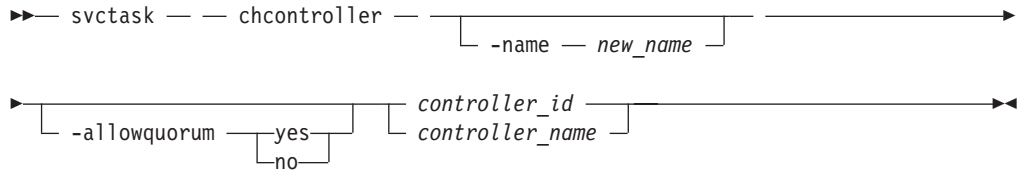
Chapter 26. Controller command

The controller command modifies the name of a storage controller.

chcontroller

The **chcontroller** command modifies the name of a controller.

Syntax



Parameters

-name *new_name*
(Optional) Specifies the new name to be assigned to the controller.

-allowquorum yes | no
(Optional) Specifies that the controller is allowed or is not allowed to support quorum disks. A value of **yes** enables a suitable controller to support quorum disks. A value of **no** disables a controller from supporting quorum disks, provided that the specified controller is not currently hosting a quorum disk.

controller_id | controller_name
(Required) Specifies the controller to modify; use either the controller name or the controller ID.

Description

This command changes the name of the controller that is specified by the *controller_id | controller_name* parameter to the value that you specify with the **-name** *new_name* parameter.

Use the optional **-allowquorum** parameter to turn quorum on or off for the specified controller.

Note: If any controller that is associated with an MDisk has the **-allowquorum** parameter set to **no**, the **setquorum** command will fail for the specified MDisk. Before setting the **-allowquorum** parameter to **yes** on any controller, check the following Web site to verify that the controller supports quorum.

www.ibm.com/storage/support/2145

You can add a new disk controller system to your SAN at any time. Follow the switch zoning guidelines in the section about switch zoning. Also, ensure that the controller is set up correctly for use with the cluster.

You can create one or more arrays on the controller using RAID-5, RAID-1, or RAID-0+1 (called RAID-10) for maximum redundancy and reliability. If your

controller provides array partitioning, you can create a single partition from the entire capacity that is available in the array. Record the LUN number that you assign to each partition. If your disk controller system requires LUN mapping, follow the mapping guidelines to map the partitions or the arrays to the cluster ports.

To add a new disk controller system to a running configuration, ensure that the cluster has detected the new storage MDisks by issuing the **svctask detectmdisk** command. The controller has automatically been assigned a default name. If you are unsure of which controller is presenting the MDisks, issue the **svcinfo lscontroller** command to list the controllers. The new controller is listed with the highest numbered default name. Record the controller name and follow the instructions in the section about determining a disk controller system name.

Give this controller a unique name by issuing the following command:

```
svctask chcontroller -name newname oldname
```

List the unmanaged MDisks by issuing the following command:

```
svcinfo lsmdisk -filtervalue mode=unmanaged:controller_name=newname
```

These MDisks correspond to the RAID arrays or partitions that you have created. Record the field controller LUN number. The field controller LUN number corresponds with the LUN number that you assigned to each of the arrays or partitions.

Create a new managed disk group and add only the RAID arrays that belong to the new controller to this MDisk group. Avoid mixing RAID types; for each set of RAID array types (for example, RAID-5 or RAID-1), create a new MDisk group. Assign this MDisk group an appropriate name; if your controller is called FAST650-abc and the MDisk group contains RAID-5 arrays, assign the MDisk a name similar to **F600-abc-R5**. Issue the following command:

```
svctask mkmdiskgrp -ext 16 -name mdisk_grp_name
-mdisk colon-separated list of RAID-x mdisks returned
```

Note: This creates a new MDisk group with an extent size of 16 MB.

An invocation example

```
svctask chcontroller -name newtwo 2
```

The resulting output

```
No feedback
```

Chapter 27. Command-line interface messages

This section lists the messages that can be displayed while you use the command-line interface (CLI).

The CLI displays a return value upon completion of the command. If the command completes normally and without error, the return code is **0**. If the command fails, the return code is **1** and the Error Code is sent to standard error. If the command succeeds, but the cluster is operating near its licensed virtualization limit, the return code can still be **1**, and a warning Error Code is sent to standard error.

When a create command is issued, the message ID that has been assigned to the new object is returned as part of the success message sent to standard output. If the **-quiet** parameter is used, only the message ID is sent to standard output.

CMMVC5000I No message was found for major rc *MAJOR_RC*, minor rc *MINOR_RC*, for action/view id *ACTION_VIEW_ID*.

Explanation

A message is missing.

Action

Contact the support center.

CMMVC5700E The parameter list is not valid.

Explanation

You have entered a list of parameters that is not supported for the command.

Action

Specify a parameter list that is supported for the command, and resubmit the command.

CMMVC5701E No object ID was specified.

Explanation

The command that you have submitted requires that you specify an object identifier name or ID number, and you did not specify an object identifier.

Action

Specify an object ID, and resubmit the command.

CMMVC5702E *VALUE* is below the minimum level.

Explanation

You entered the specified string as a value for a parameter. The parameter requires a minimum value, and the specified string is less than the required minimum value.

Action

Specify a value that is supported by the parameter, and resubmit the command.

CMMVC5703E The value or list starting with *VALUE* is above the maximum permitted for that value or has exceeded the number of items allowed in a list.

Explanation

You have entered the specified string as a value for a parameter. The string is either a standalone value or the first value in a list of values. If the string is a standalone value, the value is greater than the supported maximum value for the parameter. If the string is the first value in a list of values, the list contains more than the supported maximum number of entries for the parameter.

Action

Specify a value or list of values that is supported by the parameter, and resubmit the command.

CMMVC5704E *VALUE* is not divisible by the permitted step value.

Explanation

You have entered the specified string as a value for a parameter. The string is not a supported value for the parameter. One requirement is that the value is an even multiple of 16, and the specified string does not meet that requirement.

Action

Specify a value that is supported by the parameter, and resubmit the command.

CMMVC5705E A required parameter is missing.

Explanation

The command that you have submitted has at least one required parameter that you have not entered.

Action

Specify all of the required parameters, and resubmit the command.

CMMVC5706E An invalid argument has been entered for the *PARAMETER* parameter.

Explanation

You have entered a value for the specified parameter and the value is not supported for the parameter. The parameter supports a specific set of values.

Action

Specify a value that is supported by the parameter, and resubmit the command.

CMMVC5707E Required parameters are missing.

Explanation

The command that you have submitted has more than one required parameter that you have not entered.

Action

Specify all of the required parameters, and resubmit the command.

CMMVC5708E The *PARAMETER* parameter is missing its associated arguments.

Explanation

You have entered the specified parameter without an associated value. This parameter, like most parameters, requires an associated value.

Action

Specify the associated value, and resubmit the command.

CMMVC5709E *VALUE* is not a supported parameter.

Explanation

The specified string is not a supported parameter for the command that you have entered.

Action

Specify the correct parameter, and resubmit the command.

CMMVC5711E *VALUE* is not valid data.

Explanation

You have entered the specified string as a value for a parameter. The string is not a supported value for the parameter.

Action

Specify a value that is supported by the parameter, and resubmit the command.

CMMVC5712E Required data is missing.

Explanation

You have entered an incomplete command.

Action

Specify command completely, and resubmit the command.

CMMVC5713E Some parameters are mutually exclusive.

Explanation

Certain commands have two or more parameters that are mutually exclusive. You have submitted a command using at least two mutually exclusive parameters.

Action

Specify a supported combination of parameters, and resubmit the command.

CMMVC5714E The parameter list is empty.

Explanation

Certain parameters require one or more values in a colon separated parameter list. You have specified at least one parameter without the required parameter list.

Action

Specify at least one value for all parameters that require a value, and resubmit the command.

CMMVC5715E The parameter list does not exist.

Explanation

Certain parameters require one or more values in a colon separated parameter list. You have specified at least one parameter without the required parameter list.

Action

Specify at least one value for all parameters that require a value, and resubmit the command.

CMMVC5716E Non-numeric data was entered for the numeric field *FIELD*. Enter a numeric value.

Explanation

You have entered the specified string as a value for a parameter that supports only numeric values.

Action

Specify a numeric value in the numeric field, and resubmit the command.

CMMVC5717E No match was found for the specified unit.

Explanation

Certain parameters allow a user to specify a data unit such as mb or kb. You have entered a data unit for a parameter that supports data units, but the data unit that you have entered is not a supported data unit for the parameter.

Action

Specify the correct data unit, and resubmit the command.

CMMVC5718E An unexpected return code was received.

Explanation

The command has completed, but the acknowledgement of the command completion contains a return code that is not defined.

Action

Determine whether or not the command has succeeded. If the command has not succeeded, resubmit the command. If the problem persists, contact IBM technical support for assistance.

CMMVC5719E A value of *VALUE* requires the parameter *PARAMETER* to be specified.

Explanation

Certain commands have required combinations of parameters based on either the entry of a parameter or the value for a parameter. When you enter the specified value, you must enter the specified parameter.

Action

Specify the required parameter, and resubmit the command.

CMMVC5721E *VALUE* is not a valid time stamp format. The valid format is MMDDHHmmYYYY.

Explanation

The specified value is not a valid time-stamp format. The valid format is MMDDHHmmYYYY.

Action

Follow the correct time-stamp format, and resubmit the command.

CMMVC5722E *VALUE* is not a valid month.

Explanation

The specified value is not a valid month.

Action

Specify the correct month (MM), and resubmit the command.

CMMVC5723E *VALUE* is not a valid day.

Explanation

The specified value is not a valid day.

Action

Specify the correct day (DD), and resubmit the command.

CMMVC5724E *VALUE* is not a valid hour.

Explanation

The specified value is not a valid hour.

Action

Specify the correct hour (HH), and resubmit the command.

CMMVC5725E *VALUE* is not a valid minute.

Explanation

The specified value is not a valid minute.

Action

Specify the correct minute (mm), and resubmit the command.

CMMVC5726E *VALUE* are not valid seconds.

Explanation

The specified value are not valid seconds.

Action

Specify the correct seconds (ss), and resubmit the command.

CMMVC5727E *VALUE* is not a valid filter.

Explanation

You can filter the output of some views by using the -filtervalue parameter. The specified string that you have entered is not a supported value for the -filtervalue parameter in this view.

Action

Ensure that you use a supported value for the -filtervalue parameter, and resubmit the command.

CMMVC5728E *VALUE* should be in the format minute:hour:day:month:weekday.

Explanation

The specified value should be in the format minute:hour:day:month:weekday.

Action

Follow the correct format, and resubmit the command.

CMMVC5729E One or more components in the list is not valid.

Explanation

Certain parameters support one or more items of data in a colon separated list. At least one of the items in the list that you have entered is not correct.

Action

Ensure that you enter supported values in the list, and resubmit the command.

CMMVC5730E *VALUE* is only valid when *VALUE* has a value of *VALUE*.

Explanation

The specified command and parameter combination that you have entered requires the specified parameter value.

Action

Ensure that you specify the correct parameter value for the command and parameter combination that you enter, and resubmit the command.

CMMVC5731E *VALUE* can only be entered when *VALUE* has been entered.

Explanation

Certain commands have required combinations of parameters based either on the inclusion of a specified parameter, or on the value entered for a specified parameter. When you include the first specified string in the command, you must enter the second specified string as a parameter.

Action

Ensure that you enter a supported combination or parameters and values, and resubmit the command.

CMMVC5732E The shared-memory interface is not available, return code *RETURN_CODE*.

Explanation

You cannot submit CLI commands because the shared memory interface (SMI) is not available.

Action

Ensure that your connection to the cluster is functioning properly. If the problem persists, contact IBM technical support for assistance and report that you have received the specified message and return code.

CMMVC5733E Enter at least one parameter.

Explanation

You must specify at least one parameter for the command that you have submitted.

Action

Specify at least one parameter, and resubmit the command.

CMMVC5734E A combination of values was entered that is not valid.

Explanation

You have specified a combination of values that is not correct.

Action

Specify a supported combination of values, and resubmit the command.

CMMVC5735E The name entered is not valid. Enter an alphanumeric string that does not start with a number.

Explanation

The first character of an object name cannot be numeric.

Action

Specify an alphanumeric string that does not start with a numeric, and resubmit the command.

CMMVC5737E The parameter *PARAMETER* has been entered multiple times. Enter the parameter only one time.

Explanation

The specified parameter was entered more than once.

Action

Delete all duplicate parameters, and resubmit the command.

CMMVC5738E The argument *ARGUMENT* contains too many characters.

Explanation

The field length of the specified argument is longer than the maximum supported field length for the argument.

Action

Specify the correct argument, and resubmit the command.

CMMVC5739E The argument *ARGUMENT* does not contain enough characters.

Explanation

The field length of the specified argument is less than the minimum supported field length for the argument.

Action

Specify the correct argument, and resubmit the command.

CMMVC5740E The filter flag *VALUE* is not valid.

Explanation

You can filter the output of some views by using the -filtervalue parameter. The specified string that you have entered is not a supported value for the -filtervalue parameter in this view.

Action

Ensure that you use a supported value for the -filtervalue parameter, and resubmit the command.

CMMVC5741E The filter value *VALUE* is not valid.

Explanation

You can filter the output of some views by using the -filtervalue parameter. Each filter has an associated value. The syntax is -filtervalue filter=value. The specified string that you have entered is not a supported value for the -filtervalue filter that you specified in this view.

Action

Ensure that you use a supported value for the -filtervalue filter that you specify, and resubmit the command.

CMMVC5742E A specified parameter is out of its valid range.

Explanation

You have entered data that is not in the range of values that is supported for the parameter that you have entered.

Action

Ensure that you enter data values that are supported for the parameter that you enter, and resubmit the command.

CMMVC5743E A specified parameter does not comply with the step value.

Explanation

A parameter was specified that does not comply with the step value.

Action

Specify the correct parameter, and resubmit the command.

CMMVC5744E Too many objects were specified in the command.

Explanation

There were too many objects specified in the command.

Action

Specify the correct object, and resubmit the command.

CMMVC5745E Too few objects were specified in the request.

Explanation

There were not enough objects specified in the command.

Action

Specify the correct object, and resubmit the command.

CMMVC5746E The requested operation cannot be applied to the object specified.

Explanation

The requested operation is not valid for this object.

Action

Specify a valid operation, and resubmit the command.

CMMVC5747E The action requested is invalid - internal error.

Explanation

The operation that was requested is not valid.

Action

Specify the correct operation, and resubmit the command.

CMMVC5748E The action requested is invalid - internal error.

Explanation

The operation that was requested is not valid.

Action

Specify the correct operation, and resubmit the command.

CMMVC5749E The dump filename specified already exists.

Explanation

The dump file name that was specified already exists.

Action

Specify a different dump file name, and resubmit the command.

CMMVC5750E The dump file could not be created - the filesystem is probably full.

Explanation

The dump file was not created. The file system might be full.

Action

Not applicable.

CMMVC5751E The dump file could not be written to.

Explanation

The dump file could not be written to disk.

Action

Not applicable.

CMMVC5752E Request failed. The object contains child objects, these must be deleted first.

Explanation

The operation failed because the specified object contains child objects.

Action

Delete the child objects, and resubmit the command.

CMMVC5753E The specified object does not exist or is not a suitable candidate.

Explanation

The specified object does not exist or is not a suitable candidate.

Action

Specify the correct object, and resubmit the command.

CMMVC5754E The specified object does not exist, or the name supplied does not meet the naming rules.

Explanation

The specified object does not exist, or the name of the object does not meet the naming requirements.

Action

Specify the correct object name, and resubmit the command.

CMMVC5755E Cannot create as the sizes of the specified objects do not match.

Explanation

The sizes of the specified objects do not match.

Action

Not applicable.

CMMVC5756E Cannot perform the request as the object id is already mapped to another object or is the subject of an FC or RC relationship.

Explanation

The operation failed because the specified object is already mapped.

Action

Specify a different object, and resubmit the command.

CMMVC5757E Self Defining Structure (SDS) defaults not found - internal error.

Explanation

The defaults for the self describing structure were not found.

Action

Not applicable.

CMMVC5758E Object filename already exists.

Explanation

The object filename already exists.

Action

Specify a different object filename, and resubmit the command.

CMMVC5759E An internal error has occurred - memory could not be allocated.

Explanation

The memory cannot be allocated.

Action

Not applicable.

CMMVC5760E Failed to add the node to the cluster member list.

Explanation

The node could not be added to the cluster.

Action

Not applicable.

CMMVC5761E Failed to delete the node from the cluster member list.

Explanation

The node could not be deleted from the cluster.

Action

Not applicable.

CMMVC5762E The request did not complete before the timeout period expired.

Explanation

The operation failed because the timeout period expired.

Action

Resubmit the command.

CMMVC5763E The node failed to go online.

Explanation

The node failed to go online.

Action

Not applicable.

CMMVC5764E The mode change request is invalid - internal error

Explanation

The specified mode change is not valid.

Action

Specify a different mode, and resubmit the command.

CMMVC5765E The object specified is no longer a candidate - a change occurred during the request.

Explanation

The specified object is no longer a candidate. A change occurred during the request.

Action

Specify a different object, and resubmit the command.

CMMVC5767E One or more of the parameters specified are invalid or a parameter is missing.

Explanation

One or more of the specified parameters is not valid.

Action

Specify the correct parameter, and resubmit the command.

CMMVC5769E The requested operation requires all nodes to be online - one or more nodes are not online.

Explanation

The operation requires that all nodes be online. One or more nodes are not online.

Action

Check that each node is online, and resubmit the command.

CMMVC5770E The ssh key file supplied is invalid.

Explanation

The file for the ssh key is not valid.

Action

Specify a different file, and resubmit the command.

CMMVC5771E The operation requested could not complete, usually due to child objects existing. To force the operation, specify the force flag.

Explanation

The operation failed, probably, because the object contains child objects.

Action

Specify the -force flag to complete the operation, and resubmit the command.

CMMVC5772E The operation requested could not be performed because software upgrade is in progress.

Explanation

The operation failed because a software upgrade is in progress.

Action

Wait for the software upgrade to complete, and resubmit the command.

CMMVC5773E The object selected is in the wrong mode to perform the requested operation.

Explanation

The operation failed because the selected object is in the wrong mode.

Action

Specify the correct mode, and resubmit the command.

CMMVC5774E The userid supplied is not valid.

Explanation

The userid is not valid.

Action

Specify a different userid, and resubmit the command.

CMMVC5775E The directory attribute specified is not valid.

Explanation

The directory attribute is not valid.

Action

Specify a different directory, and resubmit the command.

CMMVC5776E The directory listing could not be retrieved.

Explanation

The directory listing could not be retrieved.

Action

Specify a different directory listing, and resubmit the command.

CMMVC5777E The node could not be added to the IO Group, because the other node in the IO Group is in the same power domain.

Explanation

The node was not added to the I/O group because the other node in the I/O Group is in the same power domain.

Action

Specify a different node from another I/O group, and resubmit the command.

CMMVC5778E Cannot create another cluster, a cluster already exists.

Explanation

The cluster was not created because one already exists.

Action

Not applicable.

CMMVC5780E The action could not be completed using the Remote Cluster name. Use the Remote Cluster Unique ID instead.

Explanation

The unique ID of the remote cluster is required for this command.

Action

Specify the unique ID of the remote cluster, and resubmit the command.

CMMVC5781E The cluster ID specified is invalid.

Explanation

The cluster ID is not valid.

Action

Specify a different cluster ID, and resubmit the command.

CMMVC5782E The object specified is offline.

Explanation

The object is offline.

Action

Specify an object that is online, and resubmit the command.

CMMVC5783E The information is not available to complete this command.

Explanation

This error is only returned when the node is in service mode.

Action

None.

CMMVC5784E The cluster name specified is not unique, specify the cluster using the cluster ID.

Explanation

The cluster name is not unique.

Action

Specify the cluster using the cluster ID, and resubmit the command.

CMMVC5785E The filename specified contains an illegal character.

Explanation

The filename contains an illegal character.

Action

Specify a valid filename, and resubmit the command.

CMMVC5786E The action failed because the cluster is not in a stable state.

Explanation

The action failed because the cluster is not in a stable state.

Action

Not applicable.

CMMVC5787E The cluster was not created because a cluster already exists.

Explanation

The cluster was not created because a cluster already exists.

Action

Not applicable.

CMMVC5788E The service IP address is not valid.

Explanation

The service IP address is not valid.

Action

Specify the correct service IP address, and resubmit the command.

CMMVC5789E The cluster was not modified because the IP address, subnet mask, service address, SNMP address, or gateway address is not valid.

Explanation

The cluster was not modified because the IP address, subnet mask, service address, SNMP address, or gateway address is not valid.

Action

Specify all correct attributes, and resubmit the command.

CMMVC5790E The node was not added to the cluster because the maximum number of nodes has been reached.

Explanation

The node was not added to the cluster because the maximum number of nodes has been reached.

Action

Not applicable.

CMMVC5791E The action failed because an object that was specified in the command does not exist.

Explanation

An entity that was specified in the command does not exist, therefore the action failed.

Action

Specify the correct entity, and resubmit the command.

CMMVC5792E The action failed because the I/O group is used for recovery.

Explanation

The action failed because the I/O group is used for recovery.

Action

Not applicable.

CMMVC5793E The node was not added to the cluster because the I/O group already contains a pair of nodes.

Explanation

The node was not added to the cluster because the I/O group already contains a pair of nodes.

Action

Not applicable.

CMMVC5794E The action failed because the node is not a member of the cluster.

Explanation

The node is not a member of the cluster, therefore the action failed.

Action

Specify a node that is contained in the cluster, and resubmit the command.

CMMVC5795E The node was not deleted because a software upgrade is in progress.

Explanation

The node was not deleted because a software upgrade is in progress.

Action

Wait for the software upgrade to complete, and resubmit the command.

CMMVC5796E The action failed because the I/O group that the node belongs to is unstable.

Explanation

A previous configuration command might not yet have completed.

Action

Wait for the previous command to complete, and resubmit the command.

CMMVC5797E The node was not deleted because this is the last node in the I/O group and there are virtual disks (VDisks) associated with the I/O group.

Explanation

The specified node is the last node in the I/O group and there are VDisks associated with the I/O group, therefore the node could not be deleted.

Action

Not applicable.

CMMVC5798E The action failed because the node is offline.

Explanation

The action failed because the node is offline.

Action

Specify a node that is online, and resubmit the command.

CMMVC5799E The shut down was not successful because there is only one online node in the I/O group.

Explanation

There is only one online node is the I/O group, therefore the shut down operation was not successful.

Action

Not applicable.

CMMVC5800E The action failed because an entity that was specified in the command does not exist.

Explanation

The entity that was specified in the command does not exist, therefore the action failed.

Action

Specify a different entity, and resubmit the command.

CMMVC5801E The upgrade of the cluster software could not proceed because every node in the cluster must be online. Either delete the node that is offline or bring the node online and resubmit the command

Explanation

The upgrade of the cluster software could not proceed because every node in the cluster must be online.

Action

Either delete the node that is offline or bring the node online, and resubmit the command.

CMMVC5802E The upgrade of the cluster software could not proceed because there is an I/O group in the cluster that contains only one node. The software upgrade requires that each node in an I/O group be shut down and restarted. If there is only one node in an I/O group, I/O operations could be lost if I/O operations are not stopped before beginning the software upgrade.

Explanation

The upgrade of the cluster software could not proceed because there is an I/O group in the cluster that contains only one node. The software upgrade requires that each node in an I/O group be shut down and restarted. If there is only one node in an I/O group, I/O operations could be lost if I/O operations are not stopped before beginning the software upgrade.

Action

Either upgrade the cluster using the -force option or specify a different node, and resubmit the command.

CMMVC5803E The entry in the error log was not marked because the error is already fixed or unfixed, or the sequence number could not be found.

Explanation

The entry in the error log was not marked because the sequence number was not found.

Action

Not applicable.

CMMVC5804E The action failed because an object that was specified in the command does not exist.

Explanation

The entity that was specified in the command does not exist, therefore the action failed.

Action

Specify a different entity, and resubmit the command.

CMMVC5805E The progress information was not returned because the FlashCopy statistics are not ready yet.

Explanation

The progress information was not returned because the FlashCopy statistics are not ready yet.

Action

Not applicable.

CMMVC5806E The action failed because an object that was specified in the command does not exist.

Explanation

The entity that was specified in the command does not exist, therefore the action failed.

Action

Specify a different entity, and resubmit the command.

CMMVC5807E The action failed because the managed disk (MDisk) cannot be changed to the specified mode.

Explanation

The action failed because the managed disk (MDisk) cannot be changed to the specified mode.

Action

Not applicable.

CMMVC5808E The action failed because the managed disk (MDisk) does not exist.

Explanation

The action failed because the managed disk (MDisk) does not exist.

Action

Specify a different MDisk, and resubmit the command.

CMMVC5809E The tracing of I/O operations was not started because it is already in progress.

Explanation

The tracing of I/O operations was not started because it is already in progress.

Action

Not applicable.

CMMVC5810E The quorum index number for the managed disk (MDisk) was not set because the MDisk is not available, either because the MDisk is offline or because it is associated with another quorum disk.

Explanation

The MDisk that you specify for this task must be online and cannot already be a quorum disk.

Action

Either change the status of the MDisk to online or specify a different MDisk, and resubmit the command.

CMMVC5811E The quorum index number for the managed disk (MDisk) was not set because the quorum disk does not exist.

Explanation

The quorum index number for the managed disk (MDisk) was not set because the quorum disk does not exist.

Action

Specify a different quorum disk, and resubmit the command.

CMMVC5812E The quorum index number for the managed disk (MDisk) was not set because the MDisk is in the wrong mode.

Explanation

The quorum index number for the managed disk (MDisk) was not set because the MDisk is not in the managed mode.

Action

- Change the mode of the MDisk, and resubmit the command
- Select an MDisk with a mode of managed, and resubmit the command

CMMVC5813E The quorum index number for the managed disk (MDisk) was not set because the MDisk has a sector size that is not valid.

Explanation

The parameter list that was specified is not valid.

Action

Specify a different sector size for the MDisk, and resubmit the command.

CMMVC5814E The quorum index number for the managed disk (MDisk) was not set because quorum is not allowed on one or more associated controllers.

Explanation

The quorum index number for the managed disk (MDisk) was not set because quorum is not allowed on one or more associated controllers.

Action

Specify an MDisk that has quorum enabled on all of its associated controllers, and resubmit the command.

CMMVC5815E The managed disk (MDisk) group was not created because an entity that was specified in the command does not exist.

Explanation

The managed disk (MDisk) group was not created because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5816E The action failed because an entity that was specified in the command does not exist.

Explanation

The action failed because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5817E The specified managed disk (MDisk) group was invalid.

Explanation

The managed disk (MDisk) group was not renamed because the name was not valid.

Action

Specify a different MDisk group name, and resubmit the command.

CMMVC5818E The managed disk (MDisk) group was not deleted because there is at least one MDisk in the group.

Explanation

The managed disk (MDisk) group was not deleted because there is at least one MDisk in the group.

Action

Not applicable.

CMMVC5819E The managed disk (MDisk) was not added to the MDisk group because the MDisk is part of another MDisk group.

Explanation

The managed disk (MDisk) was not added to the MDisk group because the MDisk is part of another MDisk group.

Action

Not applicable.

CMMVC5820E The managed disk (MDisk) was not added to the MDisk group because an entity that was specified in the command does not exist.

Explanation

The managed disk (MDisk) was not added to the MDisk group because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5821E The managed disk (MDisk) was not added to the MDisk group because not enough MDisks were included in the list.

Explanation

The managed disk (MDisk) was not added to the MDisk group because not enough MDisks were included in the list.

Action

Include more MDisks in the list, and resubmit the command.

CMMVC5822E The managed disk (MDisk) was not added to the MDisk group because too many MDisks were included in the list.

Explanation

The managed disk (MDisk) was not added to the MDisk group because too many MDisks were included in the list.

Action

Delete the extra MDisks in the list, and resubmit the command.

CMMVC5823E The managed disk (MDisk) was not deleted from the MDisk group because the MDisk is part of another MDisk group.

Explanation

The managed disk (MDisk) was not deleted from the MDisk group because the MDisk is part of another MDisk group.

Action

Not applicable.

CMMVC5824E The managed disk (MDisk) was not deleted from the MDisk group because it does not belong to the MDisk group.

Explanation

The managed disk (MDisk) was not deleted from the MDisk group because it does not belong to the MDisk group.

Action

Not applicable.

CMMVC5825E The managed disk (MDisk) was not deleted from the MDisk group because a virtual disk (VDisk) is allocated from one or more of the specified MDisks. A forced deletion is required.

Explanation

The managed disk (MDisk) was not deleted from the MDisk group because a virtual disk (VDisk) is allocated from one or more of the specified MDisks.

Action

Specify the -force option, and resubmit the command.

CMMVC5826E The virtual disk (VDisk) was not created because an entity that was specified in the command does not exist.

Explanation

The virtual disk (VDisk) was not created because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5827E The command failed as a result of either an inconsistency between two or more of the entered parameters, or an inconsistency between a parameter and the requested action.

Explanation

The command failed as a result of an inconsistency between two or more of the entered parameters.

Action

Specify one parameter, and resubmit the command.

CMMVC5828E The virtual disk (VDisk) was not created because the I/O group contains no nodes.

Explanation

The virtual disk (VDisk) was not created because the I/O group contains no nodes.

Action

Not applicable.

CMMVC5829E The image-mode or sequential-mode virtual disk (VDisk) was not created because more than one managed disk (MDisk) is specified.

Explanation

The image-mode or sequential-mode VDisk was not created because more than one MDisk is specified.

Action

Specify a different MDisk, and resubmit the command.

CMMVC5830E The image-mode virtual disk (VDisk) was not created because no managed disk (MDisk) was specified in the command.

Explanation

The image-mode virtual disk (VDisk) was not created because no managed disk (MDisk) was specified in the command.

Action

Specify a MDisk, and resubmit the command.

CMMVC5831E The virtual disk (VDisk) was not created because the preferred node for I/O operations is not part of the I/O group.

Explanation

The virtual disk (VDisk) was not created because the preferred node for I/O operations is not part of the I/O group.

Action

Specify a different node, and resubmit the command.

CMMVC5832E The property of the virtual disk (VDisk) was not modified because an entity that was specified in the command does not exist.

Explanation

The property of the virtual disk (VDisk) was not modified because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5833E The property of the virtual disk (VDisk) was not modified because there are no nodes in the I/O group.

Explanation

The property of the virtual disk (VDisk) was not modified because there are no nodes in the I/O group.

Action

Not applicable.

CMMVC5834E The I/O group for the virtual disk (VDisk) was not modified because the group is a recovery I/O group. To modify the I/O group, use the force option.

Explanation

The I/O group for the virtual disk (VDisk) was not modified because the group is a recovery I/O group.

Action

Specify the -force option, and resubmit the command.

CMMVC5835E The virtual disk (VDisk) was not expanded because an entity that was specified in the command does not exist.

Explanation

The virtual disk (VDisk) was not expanded because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5836E The virtual disk (VDisk) was not shrunk because it is locked.

Explanation

Commands might still be running in the background.

Action

Wait for all commands to complete. Use the svcinfo lsmigrate command to view any migrates running in the background.

CMMVC5837E The action failed because the virtual disk (VDisk) is part of a FlashCopy mapping.

Explanation

The action failed because the virtual disk (VDisk) is part of a FlashCopy mapping.

Action

Specify a different VDisk that is not part of a FlashCopy mapping, and resubmit the command.

CMMVC5838E The action failed because the virtual disk (VDisk) is part of a Remote Copy mapping.

Explanation

The action failed because the virtual disk (VDisk) is part of a Remote Copy mapping.

Action

Specify a different VDisk that is not part of a Remote Copy mapping, and resubmit the command.

CMMVC5839E The virtual disk (VDisk) was not shrunk because an object that was specified in the command does not exist.

Explanation

The virtual disk (VDisk) was not shrunk because an object that was specified in the command does not exist.

Action

Specify a different object, and resubmit the command.

CMMVC5840E The virtual disk (VDisk) was not deleted because it is mapped to a host or because it is part of a FlashCopy or Remote Copy mapping, or is involved in an image mode migrate.

Explanation

The virtual disk (VDisk) was not deleted because it is mapped to a host or because it is part of a FlashCopy or Metro Mirror mapping.

Action

Specify a different VDisk, and resubmit the command.

CMMVC5841E The virtual disk (VDisk) was not deleted because it does not exist.

Explanation

The virtual disk (VDisk) was not deleted because it does not exist.

Action

Specify a different VDisk, and resubmit the command.

CMMVC5842E The action failed because an object that was specified in the command does not exist.

Explanation

The action failed because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5843E The virtual disk (VDisk)-to-host mapping was not created because the VDisk does not have a capacity greater than zero bytes.

Explanation

The virtual disk (VDisk)-to-host mapping was not created because the VDisk does not have a capacity greater than zero bytes.

Action

Specify a VDisk in which its capacity is greater than zero bytes, and resubmit the command.

CMMVC5844E The virtual disk (VDisk)-to-host mapping was not created because the SCSI logical unit number (LUN) ID is not valid.

Explanation

The virtual disk (VDisk)-to-host mapping was not created because the SCSI logical unit number (LUN) ID is not valid.

Action

Specify the correct SCSI logical unit number (LUN) ID, and resubmit the command.

CMMVC5845E The extent was not migrated because an object that was specified in the command does not exist.

Explanation

The extent was not migrated because an object that was specified in the command does not exist.

Action

Specify a different object, and resubmit the command.

CMMVC5846E The virtual disk (VDisk) was not migrated because an object that was specified in the command does not exist.

Explanation

The virtual disk (VDisk) was not migrated because an object that was specified in the command does not exist.

Action

Specify a different object, and resubmit the command.

CMMVC5847E The virtual disk (VDisk) was not migrated because its associated managed disk (MDisk) is already in the MDisk group.

Explanation

The virtual disk (VDisk) was not migrated because its associated managed disk (MDisk) is already in the MDisk group.

Action

Not applicable.

CMMVC5848E The action failed because the virtual disk (VDisk) does not exist or it is being deleted.

Explanation

The action failed because the virtual disk (VDisk) does not exist or it is being deleted.

Action

Specify a different VDisk, and resubmit the command.

CMMVC5849E The migration failed because some or all of the extents are already being migrated.

Explanation

The migration failed because some or all of the extents are already being migrated.

Action

Not applicable.

CMMVC5850E The extent was not migrated because there is a problem with the source extents.

Explanation

The extent was not migrated because there is a problem with the source extents.

Action

Not applicable.

CMMVC5851E The extent was not migrated because there is a problem with the target extents.

Explanation

The extent was not migrated because there is a problem with the target extents.

Action

Not applicable.

CMMVC5852E The migration failed because there are too many migrations in progress.

Explanation

The migration failed because there are too many migrations in progress.

Action

Wait for the migration process to complete, and resubmit the command.

CMMVC5853E The action failed because there was a problem with the group.

Explanation

An attempt was made to work on a VDisk which is using an MDisk group with one of the following problems:

- The target and source MDisk groups have different extent sizes (group migrate).
- The target and source MDisk groups are the same (group migrate).
- The target and source MDisk groups are different (extents migrate).
- Target invalid group (Group Migrate).
- Source invalid group (Group Migrate).

Action

Ensure that none of the above conditions exist before reissuing the command.

CMMVC5854E The extent information was not returned because the extent is not used or does not exist.

Explanation

The extent information was not returned because the extent is not used or does not exist.

Action

Specify the correct extent, and resubmit the command.

CMMVC5855E The extent information was not returned because the managed disk (MDisk) is not used by any virtual disk (VDisk).

Explanation

The extent information was not returned because the managed disk (MDisk) is not used by any virtual disk (VDisk).

Action

Specify the correct MDisk, and resubmit the command.

CMMVC5856E The action failed because the virtual disk (VDisk) does not belong to the specified managed disk (MDisk) group.

Explanation

The action failed because the virtual disk (VDisk) does not belong to the specified managed disk (MDisk) group.

Action

Specify a different VDisk, and resubmit the command.

CMMVC5857E The action failed because the managed disk (MDisk) does not exist or it is not a member of the managed disk (MDisk) group.

Explanation

The action failed because the managed disk (MDisk) does not exist or it is not a member of the managed disk (MDisk) group.

Action

Specify a different MDisk, and resubmit the command.

CMMVC5858E The action failed because the virtual disk (VDisk) is in the wrong mode, the managed disk (MDisk) is in the wrong mode, or both are in the wrong mode.

Explanation

The action failed because the virtual disk (VDisk) is in the wrong mode, the managed disk (MDisk) is in the wrong mode, or both are in the wrong mode.

Action

Check that the VDisk and MDisk are in the correct mode, and resubmit the command.

CMMVC5859E The migration did not complete because an error occurred during the migration of the last extent on an image-mode virtual disk (VDisk).

Explanation

The migration did not complete because an error occurred during the migration of the last extent on an image-mode virtual disk (VDisk).

Action

Not applicable.

CMMVC5860E The action failed because there were not enough extents in the managed disk (MDisk) group.

Explanation

This error is also returned if a stripe set of MDisks has been specified an done or more of these MDisks does not contain enough free extents to complete the creation of the VDisk.

Action

In this case, the MDisk group reports that it has enough free capacity to create the VDisk. You can check the free capacity on each MDisk by issuing the svcinfo lsfreeextents <mdiskname/ID>. Alternatively, do not specify a stripe set and let the system choose the free extents automatically.

CMMVC5861E The action failed because there were not enough extents on the managed disk (MDisk).

Explanation

The action failed because there were not enough extents on the managed disk (MDisk).

Action

Specify another extent, and resubmit the command.

CMMVC5862E The action failed because the virtual disk (VDisk) is being formatted.

Explanation

The action failed because the virtual disk (VDisk) is being formatted.

Action

Wait for the VDisk to be successfully formatted, and resubmit the command.

CMMVC5863E The migration failed because there are not enough free extents on the target managed disk (MDisk).

Explanation

The migration failed because there are not enough free extents on the target managed disk (MDisk).

Action

Specify another free extent, and resubmit the command.

CMMVC5864E The extent information was not returned because the source extent is not used.

Explanation

The extent information was not returned because the source extent is not used.

Action

Specify a different source extent, and resubmit the command.

CMMVC5865E The action failed because the extent is out of range for the managed disk (MDisk) or virtual disk (VDisk) specified.

Explanation

The extent information was not returned because the extent is out of range for the managed disk (MDisk) or virtual disk (VDisk).

Action

Specify a different extent which is in range for the MDisk or VDisk and resubmit the command.

CMMVC5866E The action failed because the extent contains internal data.

Explanation

The extent was not migrated because the extent contains internal data.

Action

Not applicable.

CMMVC5867E The action failed because the worldwide port name is already assigned or is not valid.

Explanation

The action failed because the worldwide port name is already assigned or is not valid.

Action

Specify a different worldwide port name, and resubmit the command.

CMMVC5868E The action failed because an entity that was specified in the command does not exist.

Explanation

The action failed because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5869E The host object was not renamed because the host ID or name is not valid.

Explanation

The host object was not renamed because the host ID or name is not valid.

Action

Specify a different host ID or name, and resubmit the command.

CMMVC5870E The host object was not deleted because an entity that was specified in the command does not exist.

Explanation

The host object was not deleted because an entity that was specified in the command does not exist.

Action

Specify the correct entity, and resubmit the command.

CMMVC5871E The action failed because one or more of the configured worldwide port names is in a mapping.

Explanation

The action failed because one or more of the configured worldwide port names is in a mapping.

Action

Specify a worldwide port name that is not in a mapping, and resubmit the command.

CMMVC5872E The port (WWPN) was not added to the host object because an object that was specified in the command does not exist.

Explanation

The port (WWPN) was not added to the host object because an object that was specified in the command does not exist.

Action

Specify the correct object, and resubmit the command.

CMMVC5873E No matching WWPN.

Explanation

The action failed because there is no matching worldwide port name.

Action

Not applicable.

CMMVC5874E The action failed because the host does not exist.

Explanation

The action failed because the host does not exist.

Action

Specify a different host, and resubmit the command.

CMMVC5875E The action failed because the virtual disk (VDisk) does not exist.

Explanation

The action failed because the virtual disk (VDisk) does not exist.

Action

Specify a different VDisk, and resubmit the command.

CMMVC5876E The virtual disk (VDisk)-to-host mapping was not created because the maximum number of mappings has been reached.

Explanation

The virtual disk (VDisk)-to-host mapping was not created because the maximum number of mappings has been reached.

Action

Not applicable.

CMMVC5877E The virtual disk (VDisk)-to-host mapping was not created because the maximum number of SCSI LUNs has been allocated.

Explanation

The virtual disk (VDisk)-to-host mapping was not created because the maximum number of SCSI LUNs has been allocated.

Action

Not applicable.

CMMVC5878E The virtual disk (VDisk)-to-host mapping was not created because this VDisk is already mapped to this host.

Explanation

The virtual disk (VDisk)-to-host mapping was not created because this VDisk is already mapped to this host.

Action

Specify a different VDisk, and resubmit the command.

CMMVC5879E The virtual disk (VDisk)-to-host mapping was not created because a VDisk is already mapped to this host with this SCSI LUN.

Explanation

The virtual disk-to-host mapping was not created because this SCSI LUN is already assigned to another mapping.

Action

Specify a different SCSI LUN, and resubmit the command.

CMMVC5880E The virtual disk was not created because a capacity of zero bytes is not allowed for image mode disks.

Explanation

The virtual disk (VDisk)-to-host mapping was not created because the VDisk has a capacity of zero bytes.

Action

Specify a different VDisk, and resubmit the command.

CMMVC5881E The FlashCopy mapping was not created because an entity that was specified in the command does not exist.

Explanation

The FlashCopy mapping was not created because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5882E The FlashCopy mapping was not created because a mapping for the source or target virtual disk (VDisk) already exists.

Explanation

The FlashCopy mapping was not created because a mapping for the source or target virtual disk (VDisk) already exists.

Action

Specify a different source or target VDisk, and resubmit the command.

CMMVC5883E The FlashCopy mapping was not created because the recovery I/O group is associated with the source or target virtual disk (VDisk).

Explanation

The FlashCopy mapping was not created because the recovery I/O group is associated with the source or target virtual disk (VDisk).

Action

Specify a different recovery I/O group, and resubmit the command.

CMMVC5884E The FlashCopy mapping was not created because the source or target virtual disk (VDisk) cannot be a member of a Remote Copy mapping.

Explanation

The FlashCopy mapping was not created because the source or target virtual disk (VDisk) cannot be a member of a Remote Copy mapping.

Action

Specify a different source or target VDisk, and resubmit the command.

CMMVC5885E The FlashCopy mapping was not created because this source or target virtual disk (VDisk) cannot be a member of a FlashCopy mapping.

Explanation

The FlashCopy mapping was not created because this source or target virtual disk (VDisk) cannot be a member of a FlashCopy mapping.

Action

Specify a different source or target VDisk, and resubmit the command.

CMMVC5886E The FlashCopy mapping was not created because the source or target virtual disk (VDisk) is associated with the recovery I/O group.

Explanation

The FlashCopy mapping was not created because the source or target virtual disk (VDisk) is associated with the recovery I/O group.

Action

Specify a different source or target VDisk, and resubmit the command.

CMMVC5887E The FlashCopy mapping was not created because the source or target virtual disk (VDisk) must not be in router mode.

Explanation

The FlashCopy mapping was not created because the source or target virtual disk (VDisk) must not be in router mode.

Action

Specify a different source or target VDisk, and resubmit the command.

CMMVC5888E The action failed because an entity that was specified in the command does not exist.

Explanation

The action failed because an entity that was specified in the command does not exist.

Action

Specify the correct entity, and resubmit the command.

CMMVC5889E The FlashCopy mapping was not deleted because an entity that was specified in the command does not exist.

Explanation

The FlashCopy mapping was not deleted because an entity that was specified in the command does not exist.

Action

Specify a different entity, and resubmit the command.

CMMVC5890E The FlashCopy mapping or consistency group was not started because starting consistency group 0 is not a valid operation.

Explanation

The FlashCopy mapping or consistency group was not started because starting consistency group 0 is not a valid operation.

Action

Not applicable.

CMMVC5891E The FlashCopy consistency group was not created because the name is not valid.

Explanation

The FlashCopy consistency group was not created because the name is not valid.

Action

Specify a different name, and resubmit the command.

CMMVC5892E The FlashCopy consistency group was not created because it already exists.

Explanation

The FlashCopy consistency group was not created because it already exists.

Action

Not applicable.

CMMVC5893E The action failed because an entity that was specified in the command does not exist.

Explanation

The action failed because an entity that was specified in the command does not exist.

Action

Specify the correct entity, and resubmit the command.

CMMVC5894E The FlashCopy consistency group was not deleted because you are trying to delete consistency group 0 or the name of the consistency group is not valid.

Explanation

The FlashCopy consistency group was not deleted because the name of the consistency group is not valid or you are trying to delete consistency group 0.

Action

Specify the correct consistency group, and resubmit the command.

CMMVC5895E The FlashCopy consistency group was not deleted because it contains mappings. To delete this consistency group, a forced deletion is required.

Explanation

The FlashCopy consistency group was not deleted because it contains mappings.

Action

Specify that -force option to delete the consistency group.

CMMVC5896E The FlashCopy mapping was not deleted because the mapping or consistency group is in the preparing state. The mapping or consistency group must be stopped first.

Explanation

The FlashCopy mapping was not deleted because the mapping or consistency group is in the preparing state. The mapping or consistency group must be stopped first.

Action

Stop the consistency group, and resubmit the command.

CMMVC5897E The FlashCopy mapping was not deleted because the mapping or consistency group is in the prepared state. The mapping or consistency group must be stopped first.

Explanation

The FlashCopy mapping was not deleted because the mapping or consistency group is in the prepared state. The mapping or consistency group must be stopped first.

Action

Stop the consistency group, and resubmit the command.

CMMVC5898E The FlashCopy mapping was not deleted because the mapping or consistency group is in the copying state. The mapping or consistency group must be stopped first.

Explanation

The FlashCopy mapping was not deleted because the mapping or consistency group is in the copying state. The mapping or consistency group must be stopped first.

Action

Stop the consistency group, and resubmit the command.

CMMVC5899E The FlashCopy mapping was not deleted because the mapping or consistency group is in the stopped state. To delete the mapping, a forced deletion is required.

Explanation

The FlashCopy mapping was not deleted because the mapping or consistency group is in the stopped state.

Action

Specify the -force option to delete the mapping.

CMMVC5900E The FlashCopy mapping was not deleted because the mapping or consistency group is in the suspended state. The mapping or consistency group must be stopped first.

Explanation

The FlashCopy mapping was not deleted because the mapping or consistency group is in the suspended state. The mapping or consistency group must be stopped first.

Action

Stop the consistency group, and resubmit the command.

CMMVC5901E The FlashCopy mapping was not prepared because the mapping or consistency group is already in the preparing state.

Explanation

The FlashCopy mapping was not prepared because the mapping or consistency group is already in the preparing state.

Action

Not applicable.

CMMVC5902E The FlashCopy mapping was not prepared because the mapping or consistency group is already in the prepared state.

Explanation

The FlashCopy mapping was not prepared because the mapping or consistency group is already in the prepared state.

Action

Not applicable.

CMMVC5903E The FlashCopy mapping was not prepared because the mapping or consistency group is already in the copying state.

Explanation

The FlashCopy mapping was not prepared because the mapping or consistency group is already in the copying state.

Action

Not applicable.

CMMVC5904E The FlashCopy mapping was not prepared because the mapping or consistency group is already in the suspended state.

Explanation

The FlashCopy mapping was not prepared because the mapping or consistency group is already in the suspended state.

Action

Not applicable.

CMMVC5905E The FlashCopy mapping or consistency group was not started because the mapping or consistency group is in the idle state. The mapping or consistency group must be prepared first.

Explanation

The FlashCopy mapping or consistency group was not started because the mapping or consistency group is in the idle state.

Action

Prepare the mapping or consistency group, and resubmit the command.

CMMVC5906E The FlashCopy mapping or consistency group was not started because the mapping or consistency group is in the preparing state.

Explanation

The FlashCopy mapping or consistency group was not started because the mapping or consistency group is in the preparing state.

Action

Not applicable.

CMMVC5907E The FlashCopy mapping or consistency group was not started because the mapping or consistency group is already in the copying state.

Explanation

The FlashCopy mapping or consistency group was not started because the mapping or consistency group is already in the copying state.

Action

Not applicable.

CMMVC5908E The FlashCopy mapping or consistency group was not started because the mapping or consistency group is in the stopped state. The mapping or consistency group must be prepared first.

Explanation

The FlashCopy mapping or consistency group was not started because the mapping or consistency group is in the stopped state.

Action

Prepare the mapping or consistency group, and resubmit the command.

CMMVC5909E The FlashCopy mapping or consistency group was not started because the mapping or consistency group is in the suspended state.

Explanation

The FlashCopy mapping or consistency group was not started because the mapping or consistency group is in the suspended state.

Action

Not applicable.

CMMVC5910E The FlashCopy mapping or consistency group was not stopped because the mapping or consistency group is in the idle state.

Explanation

The FlashCopy mapping or consistency group was not stopped because the mapping or consistency group is in the idle state.

Action

Not applicable.

CMMVC5911E The FlashCopy mapping or consistency group was not stopped because the mapping or consistency group is in the preparing state.

Explanation

The FlashCopy mapping or consistency group was not stopped because the mapping or consistency group is in the preparing state.

Action

Not applicable.

CMMVC5912E The FlashCopy mapping or consistency group was not stopped because the mapping or consistency group is already in the stopped state.

Explanation

The FlashCopy mapping or consistency group was not stopped because the mapping or consistency group is already in the stopped state.

Action

Not applicable.

CMMVC5913E The properties of the FlashCopy mapping were not modified because the mapping or consistency group is in the preparing state.

Explanation

The properties of the FlashCopy mapping were not modified because the mapping or consistency group is in the preparing state.

Action

Not applicable.

CMMVC5914E The properties of the FlashCopy mapping were not modified because the mapping or consistency group is in the prepared state.

Explanation

The properties of the FlashCopy mapping were not modified because the mapping or consistency group is in the prepared state.

Action

Not applicable.

CMMVC5915E The properties of the FlashCopy mapping were not modified because the mapping or consistency group is in the copying state.

Explanation

The properties of the FlashCopy mapping were not modified because the mapping or consistency group is in the copying state.

Action

Not applicable.

CMMVC5916E The properties of the FlashCopy mapping were not modified because the mapping or consistency group is in the suspended state.

Explanation

The properties of the FlashCopy mapping were not modified because the mapping or consistency group is in the suspended state.

Action

Not applicable.

CMMVC5917E The FlashCopy mapping was not created because there is no memory in which to create the bitmap.

Explanation

The FlashCopy mapping was not created because there is no memory to create the bitmap.

Action

Not applicable.

CMMVC5918E The FlashCopy mapping was not prepared because the I/O group is offline.

Explanation

The FlashCopy mapping was not prepared because the I/O group is offline.

Action

Not applicable.

CMMVC5919E The FlashCopy mapping or consistency group was not started because the I/O group is offline.

Explanation

The FlashCopy mapping or consistency group was not started because the I/O group is offline.

Action

Not applicable.

CMMVC5920E The FlashCopy mapping was not created because the consistency group is not idle.

Explanation

The FlashCopy mapping was not created because the consistency group is not idle.

Action

Not applicable.

CMMVC5921E The properties of the FlashCopy mapping were not modified because the consistency group is not idle.

Explanation

The properties of the FlashCopy mapping were not modified because the consistency group is not idle.

Action

Not applicable.

CMMVC5922E The FlashCopy mapping was not created because the destination virtual disk (VDisk) is too small.

Explanation

The FlashCopy mapping was not created because the destination virtual disk (VDisk) is too small.

Action

Specify a different VDisk, and resubmit the command.

CMMVC5923E The FlashCopy mapping was not created because the I/O group is offline.

Explanation

The FlashCopy mapping was not created because the I/O group is offline.

Action

Not applicable.

CMMVC5924E The FlashCopy mapping was not created because the source and target virtual disks (VDisks) are different sizes.

Explanation

The FlashCopy mapping was not created because the source and target virtual disks (VDisks) are different sizes.

Action

Specify a different source and target VDisk that are the same size, and resubmit the command.

CMMVC5925E The remote cluster partnership was not created because it already exists.

Explanation

The remote cluster partnership was not created because it already exists.

Action

Specify a different remote cluster partnership, and resubmit the command.

CMMVC5926E The remote cluster partnership was not created because there are too many partnerships.

Explanation

The remote cluster partnership was not created because there are too many partnerships.

Action

Not applicable.

CMMVC5927E The action failed because the cluster ID is not valid.

Explanation

The action failed because the cluster ID is not valid.

Action

Specify the correct cluster ID, and resubmit the command.

CMMVC5928E The action failed because the cluster name is a duplicate of another cluster.

Explanation

The action failed because the cluster name is a duplicate of another cluster.

Action

Specify a different cluster name, and resubmit the command.

CMMVC5929E The Remote Copy partnership was not deleted because it has already been deleted.

Explanation

The Remote Copy partnership was not deleted because it has already been deleted.

Action

Not applicable.

CMMVC5930E The Remote Copy relationship was not created because an object that was specified in the command does not exist.

Explanation

The Remote Copy relationship was not created because an object that was specified in the command does not exist.

Action

Specify the correct object, and resubmit the command.

CMMVC5931E The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is locked.

Explanation

The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is locked.

Action

Unlock the master or auxiliary VDisk, and resubmit the command.

CMMVC5932E The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is a member of a FlashCopy mapping.

Explanation

The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is a member of a FlashCopy mapping.

Action

Not applicable.

CMMVC5933E The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is in the recovery I/O group.

Explanation

The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is in the recovery I/O group.

Action

Not applicable.

CMMVC5934E The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is in the router mode.

Explanation

The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is in the router mode.

Action

Not applicable.

CMMVC5935E The action failed because an object that was specified in the command does not exist.

Explanation

The action failed because an object that was specified in the command does not exist.

Action

Specify the correct object, and resubmit the command.

CMMVC5936E The action failed because an object that was specified in the command does not exist.

Explanation

The action failed because an object that was specified in the command does not exist.

Action

Specify the correct object, and resubmit the command.

CMMVC5937E The action failed because an object that was specified in the command does not exist.

Explanation

The action failed because an object that was specified in the command does not exist.

Action

Specify the correct object, and resubmit the command.

CMMVC5938E The Remote Copy consistency group was not deleted because the consistency group contains relationships. To delete the consistency group, the force option is required.

Explanation

Remote Copy consistency group was not deleted because the consistency group contains relationships.

Action

Specify the -force option to delete the consistency group.

CMMVC5939E The action failed because the cluster is not in a stable state.

Explanation

The action failed because the cluster is not in a stable state.

Action

Not applicable.

CMMVC5940E The cluster that contains the auxiliary virtual disk (VDisk) is unknown.

Explanation

The cluster that contains the auxiliary virtual disk (VDisk) is unknown.

Action

Not applicable.

CMMVC5941E The cluster that contains the master virtual disk (VDisk) has too many consistency groups.

Explanation

The cluster that contains the master virtual disk (VDisk) has too many consistency groups.

Action

Not applicable.

CMMVC5942E The cluster that contains the auxiliary virtual disk (VDisk) has too many consistency groups.

Explanation

The cluster that contains the auxiliary virtual disk (VDisk) has too many consistency groups.

Action

Not applicable.

CMMVC5943E The specified relationship is not valid.

Explanation

The specified relationship is not valid.

Action

Specify the correct relationship, and resubmit the command.

CMMVC5944E The specified consistency group is not valid.

Explanation

The specified consistency group is not valid.

Action

Specify the correct consistency group, and resubmit the command.

CMMVC5945E The specified master cluster is not valid.

Explanation

The specified master cluster is not valid.

Action

Specify the correct master cluster, and resubmit the command.

CMMVC5946E The specified auxiliary cluster is not valid.

Explanation

The specified auxiliary cluster is not valid.

Action

Specify the correct auxiliary cluster, and resubmit the command.

CMMVC5947E The specified master virtual disk (VDisk) is not valid.

Explanation

The specified master virtual disk (VDisk) is not valid.

Action

Specify the correct master VDisk, and resubmit the command.

CMMVC5948E The specified auxiliary virtual disk (VDisk) is not valid.

Explanation

The specified auxiliary virtual disk (VDisk) is not valid.

Action

Specify the auxiliary VDisk, and resubmit the command.

CMMVC5949E The specified relationship is unknown.

Explanation

The specified relationship is unknown.

Action

Specify a different relationship, and resubmit the command.

CMMVC5950E The specified consistency group is unknown.

Explanation

The specified consistency group is unknown.

Action

Specify a different consistency group, and resubmit the command.

CMMVC5951E The operation cannot be performed because the relationship is not a stand-alone relationship.

Explanation

The operation cannot be performed because the relationship is not a stand-alone one.

Action

Not applicable.

CMMVC5952E The relationship and consistency group have different master clusters.

Explanation

The relationship and consistency group have different master clusters.

Action

Not applicable.

CMMVC5953E The relationship and group have different auxiliary clusters.

Explanation

The relationship and group have different auxiliary clusters.

Action

Not applicable.

CMMVC5954E The master and auxiliary virtual disks (VDisks) are different sizes.

Explanation

The master and auxiliary virtual disks (VDisks) are different sizes

Action

Not applicable.

CMMVC5955E The maximum number of relationships has been reached.

Explanation

The maximum number of relationships has been reached.

Action

Not applicable.

CMMVC5956E The maximum number of consistency groups has been reached.

Explanation

The maximum number of consistency groups has been reached.

Action

Not applicable.

CMMVC5957E The master virtual disk (VDisk) is already in a relationship.

Explanation

The master virtual disk (VDisk) is already in a relationship.

Action

Specify a different master VDisk, and resubmit the command.

CMMVC5958E The auxiliary virtual disk (VDisk) is already in a relationship.

Explanation

The auxiliary virtual disk (VDisk) is already in a relationship.

Action

Specify a different auxiliary VDisk, and resubmit the command.

CMMVC5959E There is a relationship that already has this name on the master cluster.

Explanation

There is a relationship that already has this name on the master cluster.

Action

Specify a different name, and resubmit the command.

CMMVC5960E There is a relationship that already has this name on the auxiliary cluster.

Explanation

There is a relationship that already has this name on the auxiliary cluster.

Action

Specify a different name, and resubmit the command.

CMMVC5961E There is a consistency group that already has this name on the master cluster.

Explanation

There is a consistency group that already has this name on the master cluster.

Action

Specify a different name, and resubmit the command.

CMMVC5962E There is a consistency group that already has this name on the auxiliary cluster.

Explanation

There is a consistency group that already has this name on the auxiliary cluster.

Action

Specify a different name, and resubmit the command.

CMMVC5963E No direction has been defined.

Explanation

No direction has been defined.

Action

Not applicable.

CMMVC5964E The copy priority is not valid.

Explanation

The copy priority is not valid.

Action

Not applicable.

CMMVC5965E The virtual disks (VDisks) are in different I/O groups on the local cluster.

Explanation

The virtual disks (VDisks) are in different I/O groups on the local cluster.

Action

Not applicable.

CMMVC5966E The master virtual disk (VDisk) is unknown.

Explanation

The master virtual disk (VDisk) is unknown.

Action

Specify a different master VDisk, and resubmit the command.

CMMVC5967E The auxiliary virtual disk (VDisk) is unknown.

Explanation

The auxiliary virtual disk (VDisk) is unknown.

Action

Specify a different auxiliary VDisk, and resubmit the command.

CMMVC5968E The relationship cannot be added because the states of the relationship and the consistency group do not match.

Explanation

The relationship cannot be added because the states of the relationship and the consistency group do not match.

Action

Not applicable.

CMMVC5969E The Remote Copy relationship was not created because the I/O group is offline.

Explanation

The Remote Copy relationship was not created because the I/O group is offline.

Action

Not applicable.

CMMVC5970E The Remote Copy relationship was not created because there is not enough memory.

Explanation

The Remote Copy relationship was not created because there is not enough memory.

Action

Not applicable.

CMMVC5971E The operation was not performed because the consistency group contains no relationships.

Explanation

The operation was not performed because the consistency group contains no relationships.

Action

Not applicable.

CMMVC5972E The operation was not performed because the consistency group contains relationships.

Explanation

The operation was not performed because the consistency group contains relationships.

Action

Not applicable.

CMMVC5973E The operation was not performed because the consistency group is not synchronized.

Explanation

The operation was not performed because the consistency group is not synchronized.

Action

Specify the Force option when starting the consistency group.

CMMVC5974E The operation was not performed because the consistency group is offline.

Explanation

The operation was not performed because the consistency group is offline.

Action

Not applicable.

CMMVC5975E The operation was not performed because the cluster partnership is not connected.

Explanation

The operation was not performed because the cluster partnership is not connected.

Action

Not applicable.

CMMVC5976E The operation was not performed because the consistency group is in the freezing state.

Explanation

The operation was not performed because the consistency group is in the freezing state.

Action

Not applicable.

CMMVC5977E The operation was not performed because it is not valid given the current consistency group state.

Explanation

The operation was not performed because it is not valid given the current consistency group state.

Action

Not applicable.

CMMVC5978E The operation was not performed because the relationship is not synchronized.

Explanation

The operation was not performed because the relationship is not synchronized.

Action

Not applicable.

CMMVC5980E The operation was not performed because the master and auxiliary clusters are not connected.

Explanation

The operation was not performed because the master and auxiliary clusters are not connected.

Action

Not applicable.

CMMVC5981E The operation was not performed because the relationship is in the freezing state.

Explanation

The operation was not performed because the relationship is in the freezing state.

Action

Not applicable.

CMMVC5982E The operation was not performed because it is not valid given the current relationship state.

Explanation

The operation was not performed because it is not valid given the current relationship state.

Action

Not applicable.

CMMVC5983E dump file was not created. This may be due to the file system being full.

Explanation

dump file was not created. This may be due to the file system being full.

Action

Not applicable.

CMMVC5984E The dump file was not written to disk. This may be due to the file system being full.

Explanation

The dump file was not written to disk. This may be due to the file system being full.

Action

Not applicable.

CMMVC5985E The action failed because the directory that was specified was not one of the following directories: /dumps, /dumps/iostats, /dumps/iotrace, /dumps/feature, /dumps/configs, /dumps/elogs, or /home/admin/upgrade.

Explanation

The action failed because the directory that was specified was not one of the following directories:

- /dumps
- /dumps/iostats
- /dumps/iotrace
- /dumps/feature
- /dumps/configs
- /dumps/elogs
- /home/admin/upgrade

Action

Specify one of the above directories, and resubmit the command.

CMMVC5986E The tracing of I/O operations was not started because the virtual disk (VDisk) or managed disk (MDisk) failed to return any statistics.

Explanation

The tracing of I/O operations was not started because the virtual disk (VDisk) or managed disk (MDisk) failed to return statistics.

Action

Not applicable.

CMMVC5987E *VALUE* is not a valid command line option.

Explanation

The specified string that you have entered is not a supported command line option.

Action

Specify a supported option, and resubmit the command.

CMMVC5988E command should not be run by the root userid. Use the admin userid.

Explanation

This command should not be issued if you are logged in with a root user ID. Use the admin userid.

Action

Log off of the root user ID and log in as admin.

CMMVC5989E The operation was not performed because the relationship is offline.

Explanation

The operation was not performed because the relationship is offline.

Action

Not applicable.

CMMVC5990E The FlashCopy consistency group was not stopped as there are no FlashCopy mappings within the group.

Explanation

The FlashCopy consistency group was not stopped as there are no FlashCopy mappings within the group.

Action

Not applicable.

CMMVC5991E The Remote Copy consistency group was not stopped as there are no Remote Copy relationships within the group.

Explanation

The Remote Copy consistency group was not stopped as there are no Remote Copy relationships within the group.

Action

Not applicable.

CMMVC5992E The Remote Copy consistency group was not stopped as there are no Remote Copy relationships within the group.

Explanation

The Remote Copy consistency group was not stopped as there are no Remote Copy relationships within the group.

Action

Not applicable.

CMMVC5993E The specified upgrade package does not exist.

Explanation

The specified upgrade package does not exist.

Action

Not applicable.

CMMVC5994E Error in verifying the signature of the upgrade package.

Explanation

The system could not verify the signature of the upgrade package due to the following reasons:

- There is not enough space on the system to copy the file.
- The package is incomplete or contains errors.

Action

If the copy failed with an error indicating that there was insufficient space on the system, free up additional space on your system. Otherwise, ensure that the cluster time and date stamp on the signature is correct. (For example, the time and date cannot be in the future.)

CMMVC5995E Error in unpacking the upgrade package.

Explanation

The most likely cause of this error is lack of system space.

Action

Reboot the node and unpack the upgrade package again.

CMMVC5996E The specific upgrade package cannot be installed over the current version.

Explanation

The upgrade package is not compatible with the current version or the system.

Action

Check the available upgrade packages and find the correct upgrade package for your current version and for your system. If the upgrade package is correct for your system, check the version requirements for the package. You might have to upgrade the current version to an intermediate version before you upgrade to the latest version. (For example, if your current version is 1 and you are trying to upgrade to version 3, you might need to upgrade to version 2 before applying the version 3 upgrade.)

CMMVC5999W Featurization for this facility has not been enabled.

Explanation

Featurization for this facility has not been enabled.

Action

Not applicable.

CMMVC6000W Featurization for this facility has not been enabled.

Explanation

Featurization for this facility has not been enabled.

Action

Not applicable.

CMMVC6001E The FlashCopy consistency group was not started as there are no FlashCopy mappings within the group.

Explanation

The FlashCopy consistency group was not started as there are no FlashCopy mappings within the group.

Action

Create a FlashCopy within the appropriate group.

CMMVC6002E This command can only be run on a node that is in service mode.

Explanation

This command can only be run on a node that is in service mode.

Action

Not applicable.

CMMVC6003E This command can not be run on a node that is in service mode.

Explanation

This command can not be run on a node that is in service mode.

Action

Not applicable.

CMMVC6004E The delimiter value *VALUE* is invalid.

Explanation

The specified value is not a valid delimiter value.

Action

Specify a different delimiter.

CMMVC6005E The view request failed as the specified object is not a member of an appropriate group.

Explanation

A view was request on an object that has been incorrectly initialized.

Action

Ensure that the object is correctly initialized before resubmitting the view request.

CMMVC6006E The managed disk (MDisk) was not deleted because the resource was busy.

Explanation

An attempt was made to delete an MDisk from a MDisk group that is being used as a source and destination for migration operations.

Action

Ensure that the MDisk group is not being used for migration operations before reissuing the command.

CMMVC6007E The two passwords that were entered do not match.

Explanation

The two passwords entered for verification of your password change were not the same.

Action

Re-enter the passwords.

CMMVC6008E The key already exists.

Explanation

An attempt was made to load a duplicate SSH key.

Action

Not applicable.

CMMVC6009E Unable to malloc a block of memory in which to copy the returned data.

Explanation

The command line was unable to allocate a block of memory in which to copy the results of the query.

Action

Resubmit the command. If the problem persists, contact IBM technical support for assistance.

CMMVC6010E Unable to complete the command as there are insufficient free extents, or the command requested an expansion of 0 size.

Explanation

There are not enough free extents to meet the request.

Action

Not applicable.

CMMVC6011E This cluster is part of a remote cluster partnership. Because this upgrade pacjkage will make changes to the cluster state, it cannot be applied to the current code level until all remote cluster partnerships are deleted.

Explanation

You have attempted to apply software when a Remote Copy relationship to a remote cluster exists.

Action

Delete the Remote Copy relationship to the remote clusters, and resubmit the command.

CMMVC6012W The virtualized storage capacity is approaching the amount that you are licensed to use.

Explanation

The requested action has completed. However, the limits permitted by the license you purchased are approaching.

Action

Subsequent actions might require that you increase your licensed limits.

CMMVC6013E The command failed because there is a consistency group mismatch on the aux cluster.

Explanation

The action has failed as there was a difference in attributes between the Metro Mirror consistency groups involved.

Action

Ensure that the attributes of the two Metro Mirror consistency groups match before resubmitting the command.

CMMVC6014E The command failed because the requested object is either unavailable or does not exist.

Explanation

The command failed because the requested object is either unavailable or does not exist.

Action

Ensure that all parameters have been correctly entered. If this is the case the determine why the object is unavailable, then resubmit the command.

CMMVC6015E A delete request is already in progress for this object.

Explanation

A delete request is already in progress for this object.

Action

Not applicable.

CMMVC6016E The action failed as there would be, or are, no more disks in the MDisk group.

Explanation

The action failed as there would be, or are, no more disks in the I/O group.

Action

Ensure that all parameters have been correctly entered.

CMMVC6017E A parameter or argument contains invalid characters. Ensure that all characters are ASCII.

Explanation

The command-line interface (CLI) will only accept ASCII input.

Action

Ensure that all input to the CLI is ASCII, then resubmit the command.

CMMVC6018E The software upgrade pre-install process failed.

Explanation

The software upgrade failed as there was an error during the preprocessing. The package is either invalid or corrupted.

Action

Ensure the package is a valid upgrade package. Download the package from the source location again as it might have been corrupted during a network transfer.

CMMVC6019E The software upgrade failed as a node pended as the upgrade was in progress.

Explanation

The software upgrade failed as a node pended as the upgrade was in progress.

Action

Ensure that all nodes are online and available before restarting the upgrade process.

CMMVC6020E The software upgrade failed as the system was unable to distribute the software package to all nodes.

Explanation

The software upgrade failed as the system was unable to distribute the software package to all nodes.

Action

Ensure that all nodes are correctly zoned and that all nodes are online and can see the other nodes in the cluster. You might also want to check the error log.

CMMVC6021E The system is currently busy performing another request. Try again later.

Explanation

The requested action failed as the system is processing another request.

Action

Wait before resubmitting the request.

CMMVC6022E The system is currently busy performing another request. Try again later.

Explanation

The requested action failed as the system is processing another request.

Action

Wait before resubmitting the request.

CMMVC6023E The system is currently busy performing another request. Try again later.

Explanation

The requested action failed as the system is processing another request.

Action

Wait before resubmitting the request.

CMMVC6024E The auxiliary VDisk entered is invalid.

Explanation

The auxiliary VDisk is entered as a parameter in the command-line interface is not a valid auxiliary VDisk.

Action

Select a valid auxiliary VDisk, and resubmit the command.

CMMVC6025E The RC consistency group Master cluster is not the local cluster.

Explanation

The auxiliary VDisk is entered as a parameter in the command-line interface is not a valid auxiliary VDisk.

Action

Resubmit the command with a consistency group that belongs to the local cluster.

CMMVC6026E The RC consistency group is not in the stopped state.

Explanation

The action failed as the Metro Mirror consistency group is not in the stopped state.

Action

Ensure that the Metro Mirror consistency group is in the stopped state before resubmitting the command.

CMMVC6027E The RC consistency group is not the primary master.

Explanation

The RC consistency group requested in the command is not the Metro Mirror primary master.

Action

Ensure that the parameters have been entered correctly on the command line.

CMMVC6028E This upgrade package cannot be applied to the current software level because it contains changes to the cluster state and there are remote cluster partnership defined.

Explanation

The action failed because there is a connected remote cluster. The upgrade cannot be applied because it would render the remote cluster at a different code level to the remote cluster.

Action

Ensure that the cluster partnership is deconfigured before resubmitting the command. Ensure that you deconfigure the remote cluster and upgrade the code on it before reconfiguring the cluster partnership.

CMMVC6029E All nodes must have identical code level before a concurrent code upgrade can be performed.

Explanation

The concurrent upgrade failed as two or more nodes were at differing code levels. All nodes must be at the same code level before a software upgrade can be performed.

Action

Use the service mode to bring all nodes to the same level before resubmitting the concurrent upgrade.

CMMVC6030E The operation was not preformed because the FlashCopy mapping is part of a consistency group. The action must be performed at the consistency group level.

Explanation

An attempt was made to stop a FlashCopy mapping. This failed as the FlashCopy mapping is part of a consistency group.

Action

Issue the stop command to the FlashCopy consistency group. This will stop all FlashCopies within that group that are in progress.

CMMVC6031E The operation was not performed because the FlashCopy consistency group is empty.

Explanation

An attempt was made to prestart an empty FlashCopy consistency group.

Action

Not applicable.

CMMVC6032E The operation was not performed because one or more of the entered parameters is invalid for this operation.

Explanation

An invalid parameter was entered for the command.

Action

If attempting to change the I/O group to which the VDisk belongs, ensure that the VDisk is not already a part of the group.

CMMVC6033E The action failed due to an internal error.

Explanation

An internal error caused the action to fail.

Action

Not applicable.

CMMVC6034E The action failed because the maximum number of objects has been reached.

Explanation

The action failed because the maximum number of objects has been reached.

Action

Not applicable.

CMMVC6035E The action failed as the object already exists.

Explanation

An operation was requested to create an object that already exists.

Action

Ensure that the name you are attempting to apply to a new object does not exist, or change the name before re-issuing the command.

CMMVC6036E An invalid action was requested.

Explanation

The action failed because it is not a valid action with the command that was issued.

Action

Issue an action that is valid with the command.

CMMVC6037E The action failed as the object is not empty.

Explanation

The action failed because an object was specified.

Action

Resubmit the command without specifying an object.

CMMVC6038E The action failed as the object is empty.

Explanation

The action failed because an object was not specified.

Action

Specify an object, and resubmit the command.

CMMVC6039E The action failed as the object is not a member of a group.

Explanation

The action failed because the object is not a member of a group.

Action

Specify an object that is part of a group, and resubmit the command.

CMMVC6040E The action failed as the object is not a parent.

Explanation

The action failed because the object is not a parent object.

Action

Specify an object that is a parent, and resubmit the command.

CMMVC6041E The action failed as the cluster is full.

Explanation

The action failed because the cluster is full.

Action

Remove data from the cluster, and resubmit the command.

CMMVC6042E The action failed as the object is not a cluster member.

Explanation

The action failed because the object is not a member of the cluster.

Action

Specify an object that is a member of the cluster, and resubmit the command.

CMMVC6043E The action failed as the object is a member of a group.

Explanation

The action failed because the object is a member of a group.

Action

Specify an object that is not a member of a group, and resubmit the command.

CMMVC6044E The action failed as the object is a parent.

Explanation

The action failed because the object is a parent object.

Action

Specify an object that is not a parent object, and resubmit the command.

CMMVC6045E The action failed, as the -force flag was not entered.

Explanation

The action failed because the -force option was not entered.

Action

Specify the -force option in the command.

CMMVC6046E The action failed as too many candidates were selected.

Explanation

The action failed because too many candidates were specified.

Action

Specify fewer candidates in the command.

CMMVC6047E The action failed as too few candidates were selected.

Explanation

An action was requested with too few candidate objects.

Action

Determine the correct number of candidates required for the specific command and re-issue the command.

CMMVC6048E The action failed as the object is busy.

Explanation

The action failed because the object is busy.

Action

Not applicable.

CMMVC6049E The action failed as the object is not ready.

Explanation

The action failed because the object is not ready.

Action

Not applicable.

CMMVC6050E The action failed as the command was busy.

Explanation

The action failed because the command is busy.

Action

Not applicable.

CMMVC6051E An unsupported action was selected.

Explanation

The action failed because it is not valid with the command.

Action

Specify an action that is valid with the command.

CMMVC6052E The action failed as the object is a member of a FlashCopy mapping.

Explanation

The object is a member of a FlashCopy mapping, thus it cannot be deleted.

Action

Specify an object that is not a member of a FlashCopy mapping, or remove the object from the FlashCopy mapping.

CMMVC6053E An invalid WWPN was entered.

Explanation

An invalid worldwide port name (WWPN) was specified.

Action

Specify a valid WWPN.

CMMVC6054E The action failed as not all nodes are online.

Explanation

One or more nodes are not online.

Action

Check that each node is online, and resubmit the command.

CMMVC6055E The action failed as an upgrade is in progress.

Explanation

The action failed because a software upgrade is in progress.

Action

Wait for the software upgrade to complete, and resubmit the command.

CMMVC6056E The action failed as the object is too small.

Explanation

The action failed because the object is too small.

Action

Specify a different object, and resubmit the command.

CMMVC6057E The action failed as the object is the target of a FlashCopy mapping.

Explanation

The object is the target of a FlashCopy mapping, thus it cannot be deleted.

Action

Specify an object that is not the target of a FlashCopy mapping, or remove the object from the FlashCopy mapping.

CMMVC6058E The action failed as the object is in the recovery HWS.

Explanation

An attempt was made to perform an operation on a node that is in the recovery IO group.

Action

Get the node into one of the other IO Groups and re-issue the command.

CMMVC6059E The action failed as the object is in an invalid mode.

Explanation

The action failed because the object is in the wrong mode.

Action

Check that the object is in the correct mode, and resubmit the command.

CMMVC6060E The action failed as the object is being deleted.

Explanation

The action failed because the object is being deleted.

Action

Not applicable.

CMMVC6061E The action failed as the object is being resized.

Explanation

The action failed because the object is being resized.

Action

Check that the object is in the correct mode, and resubmit the command.

CMMVC6062E The action failed as the object is being moved between HWS.

Explanation

An attempt was made to perform an action against an object that is currently being moved between IO groups.

Action

Re-issue the command when the move operation has completed.

CMMVC6063E The action failed as there are no more disks in the group.

Explanation

An attempt was made to perform an action against a group that contained no disks.

Action

Either add disks to the group and re-issue the command, or select another group against which to execute the action.

CMMVC6064E The action failed as the object has an invalid name.

Explanation

An attempt was made to create or rename an object using an invalid name.

Action

Use a name that meets the naming standards and re-issue the command.

CMMVC6065E The action failed as the object is not in a group.

Explanation

An attempt was made to perform an action on an object that was not in an appropriate group.

Action

Ensure that the object is a member of an appropriate group and re-issue the command.

CMMVC6066E The action failed as the system is running low on memory.

Explanation

The system is running low on memory.

Action

Not applicable.

CMMVC6067E The action failed as the SSH key was not found.

Explanation

An attempt was made to perform an action using an SSH key that does not exist.

Action

Re-issue the command using a key that does exist.

CMMVC6068E The action failed as there are no free SSH keys.

Explanation

An attempt was made to use an SSH key when there are no free SSH keys.

Action

Upload additional keys and re-issue the command.

CMMVC6069E The action failed as the SSH key is already registered.

Explanation

An attempt was made to register an SSH key that was already registered.

Action

Not applicable.

CMMVC6070E An invalid or duplicated parameter, unaccompanied argument, or incorrect argument sequence has been detected. Ensure that the input is as per the help.

Explanation

The parameters entered for a command were invalid.

Action

Correct the parameters and re-issue the command.

CMMVC6071E The virtual disk (VDisk)-to-host mapping was not created because the VDisk is already mapped to a host.

Explanation

The virtual disk is already mapped to a host.

Action

Not applicable.

CMMVC6073E The maximum number of files has been exceeded.

Explanation

The maximum number of files has been exceeded.

Action

Not applicable.

CMMVC6074E The command failed as the extent has already been assigned.

Explanation

The command failed as the extent has already been assigned.

Action

Assign a different extent, and resubmit the command.

CMMVC6075E The expand failed as the last extent is not a complete extent.

Explanation

The expand failed as the last extent is not a complete extent.

Action

Assign a different extent, and resubmit the command.

CMMVC6076E The command failed because the virtual disk cache is not empty. Either wait for the cache to flush or use the force flag to discard the contents of the cache.

Explanation

The command failed due to an error during the flushing of the VDisk.

Action

Not applicable.

CMMVC6077E WARNING - Unfixed errors should be fixed before applying software upgrade. Depending on the nature of the errors, they might cause the upgrade process to fail. It is highly recommended to fix these errors before proceeding. If a particular error cannot be fixed, contact the support center.

Explanation

Unfixed errors should be fixed before applying software upgrade. Depending on the nature of the errors, they might cause the upgrade process to fail. It is highly recommended to fix these errors before proceeding.

Action

If the error cannot be fixed, contact the support center.

CMMVC6078E The action failed because the object is in an invalid mode.

Explanation

An attempt was made to perform an action against an object in a mode that did not allow for that action to be performed.

Action

Get the object into a suitable mode and re-issue the command.

CMMVC6079E Metatdata recovery could not complete the operation because a parameter is invalid.

Explanation

Metatdata recovery could not complete the operation because a parameter is invalid.

CMMVC6081E Metadata Recovery is busy processing the previous operation.

Explanation

Metadata Recovery is busy processing the previous operation.

CMMVC6082E The attempt to abort metadata recovery failed because the previous operation has completed.

Explanation

The attempt to abort metadata recovery failed because the previous operation has completed.

Action

None.

CMMVC6083E Metadata recovery could not find a valid dumpfile required for the rebuild operation.

Explanation

Metadata recovery could not find a valid dumpfile required for the rebuild operation.

CMMVC6084E Metadata recovery could not create/open/write the scan file, the disk might be full.

Explanation

Metadata recovery could not create/open/write the scan file, the disk might be full.

CMMVC6085E Metadata recovery could not create/open/write the dump file, the disk might be full.

Explanation

Metadata recovery could not create/open/write the dump file, the disk might be full.

CMMVC6086E Metadata recovery could not create/open/write the progress file, the disk might be full.

Explanation

Metadata recovery could not create/open/write the progress file, the disk might be full.

CMMVC6087E Metadata recovery could not map the buffers necessary to complete the operation.

Explanation

Metadata recovery could not map the buffers necessary to complete the operation.

CMMVC6088E The lba at which metadata recovery was requested does not contain metadata.

Explanation

The lba at which metadata recovery was requested does not contain metadata.

CMMVC6089E The metadata at the requested lba is flagged as invalid.

Explanation

The metadata at the requested lba is flagged as invalid.

CMMVC6090E The metadata header checksum verification failed.

Explanation

The metadata header checksum verification failed.

CMMVC6091E The metadata region checksum verification failed.

Explanation

The metadata region checksum verification failed.

CMMVC6092E The metadata recovery operation was aborted.

Explanation

The metadata recovery operation was aborted.

CMMVC6093E Metadata recovery internal error - (read only)

Explanation

Metadata recovery internal error - (read only)

CMMVC6095E Metadata recovery encountered the end of the disk.

Explanation

Metadata recovery encountered the end of the disk.

CMMVC6096E Metadata recovery encountered an error from a lower layer - (vl no resource).

Explanation

Metadata recovery encountered an error from a lower layer - (vl no resource).

CMMVC6097E Metadata recovery encountered an error from a lower layer - (vl failure).

Explanation

Metadata recovery encountered an error from a lower layer - (vl failure).

CMMVC6098E The copy failed as the specified node is the configuration node.

Explanation

The copy failed because the specified node is the configuration node.

Action

Not applicable.

CMMVC6100E *OPTION* not consistent with *ACTION*

Explanation

The specified option is not supported for the specified action.

Action

Remove the option, and resubmit the command.

CMMVC6101E *OPTION* not consistent with *OPTION*

Explanation

The two specified options cannot be used together.

Action

Remove one of the options, and resubmit the command.

CMMVC6102E *OPTION* and *OPTION* are alternatives

Explanation

The two specified options are alternatives, and cannot be used together.

Action

Remove one of the options, and resubmit the command.

CMMVC6103E Problem with *FILENAME*: *DETAILS*

Explanation

A problem occurred when opening the specified file. Determine the cause of the problem and correct it before trying again.

Action

Correct the problem, and resubmit the command.

CMMVC6104E Action *ACTION* not run

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6105E Different names for source *SOURCE_CLUSTER_NAME* and target *TARGET_CLUSTER_NAME* clusters

Explanation

The backup configuration cannot be restored to the target cluster because the source and target cluster have different names.

Action

Perform one of the following actions: (1) Use a different backup configuration. (2) Delete the cluster and recreate it with the same name as that stored in the backup configuration file.

CMMVC6106W Target cluster has non-default id_alias *ALIAS*.

Explanation

The specified id_alias of the target cluster is a non-default value. Clusters should have the default value. The non-default value suggests that the cluster is customized and is not suitable for restoration. Restoration changes the id_alias.

Action

Change the id_alias to a default value, and resubmit the command.

CMMVC6107E *NUMBER_OF_OBJECTS* io_grp objects in target cluster; *NUMBER_OF_REQUIRED_OBJECTS* are required

Explanation

The number of I/O groups in the target cluster is not sufficient to accommodate the I/O groups defined in the backup configuration file. Determine why there are not enough I/O groups.

Action

Correct the problem, and resubmit the command.

CMMVC6108I Disk controller system with a WWNN of *WWNN_VALUE* found.

Explanation

A disk controller system with the required WWNN has been found.

Action

Not applicable.

CMMVC6109E Disk controller system with a WWNN of *WWNN_VALUE* not available.

Explanation

A disk controller system with the specified WWNN has been found. Ensure that the specified disk controller system is available to the cluster.

Action

Ensure that the required disk controller system is available to the cluster, and resubmit the command.

CMMVC6110E Bad code level: *VALUE*.

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6111E The cluster code level could not be determined from *VALUE*.

Explanation

The code level of the cluster could not be determined. The code level should be of the format x.y.z, where x, y, and z are integers.

Action

If the cause of the problem cannot be determined, contact IBM technical support for assistance.

CMMVC6112W *OBJECT_TYPE OBJECT_NAME* has a default name.

Explanation

An object in the cluster has a default name. This can cause problems when restoring a cluster because default names are changed during restoration. Object IDs are also changed during restoration.

Action

Choose an appropriate name for each object in the cluster, and resubmit the command.

CMMVC6113E The command *COMMAND* has failed with return code *RETURN_CODE*.

Explanation

An attempt to run a command remotely failed using secure communications.

Action

Determine the cause of the problem, and resubmit the command.

CMMVC6114E No help for action *ACTION*.

Explanation

There is no help for the specified action topic.

Action

Not applicable.

CMMVC6115W Feature *FEATURE_PROPERTY* mismatch: *VALUE* expected; *VALUE* found.

Explanation

The features in the backup configuration file and the target cluster do not match. There should be an exact match between the two. Nevertheless, the restore of the configuration can continue.

Action

Not applicable.

CMMVC6116I Feature match for *FEATURE*.

Explanation

The features in the backup configuration file and the target cluster are an exact match.

Action

Not applicable.

CMMVC6117E *FIX_OR_FEATURE* is not available.

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6118I *TYPE* with *PROPERTY PROPERTY_VALUE* and *PROPERTY PROPERTY_VALUE* found.

Explanation

An object in the cluster has been found with the correct properties.

Action

Not applicable.

CMMVC6119E *TYPE* with *PROPERTY PROPERTY_VALUE* not found.

Explanation

An object in the cluster with the correct properties has not been found. Restoration cannot proceed without the object.

Action

Determine why the object cannot be found. Ensure that the object is available, and resubmit the command.

CMMVC6120E Target is not the configuration node

Explanation

The target is not the configuration node.

Action

Redirect the action against the configuration node, and resubmit the command.

CMMVC6121E No cluster id or id_alias in backup configuration.

Explanation

Neither the cluster id_alias nor the ID can be extracted from the backup configuration file.

Action

If the cause of the problem cannot be determined, contact IBM technical support for assistance.

CMMVC6122E No *TYPE* with *PROPERTY VALUE* is present in the table.

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6123E No *PROPERTY* for *TYPE NAME*.

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6124E No *TYPE* with *PROPERTY VALUE*

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6125E No unique ID for *TYPE NAME*

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6126E No *TYPE* with unique ID *VALUE*

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6127I The SSH key *IDENTIFIER* for *USER* is already defined; the SSH key will not be restored

Explanation

An identical SSH key for this user is already defined on the cluster. Therefore, the key in the backup file will not be restored.

Action

Specify a different SSH key, and resubmit the command.

CMMVC6128W *DIRECTORY*

Explanation

The files in the specified directory cannot be listed.

Action

Determine why the files cannot be listed, correct the problem, and resubmit the command.

CMMVC6129E VDisk-to-host map objects have VDisk_UID values that are not consistent

Explanation

All of the VDisk-to-host map objects do not have the same number for the VDisk LUN instance. Therefore, there is a possibility the backup configuration file is corrupt. The LUN instance number should be the same for all VDisk-to-host map objects that are associated with a specific VDisk. The LUN instance number is incorporated into the VDisk_UID property.

Action

Determine why the LUN instance number is not the same, correct the problem, and resubmit the command.

CMMVC6130W Inter-cluster *PROPERTY VALUE* will not be restored.

Explanation

The restoration of inter-cluster objects is not supported.

Action

Not applicable.

CMMVC6131E No location cluster information

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6132E The object *OBJECT* of type *TYPE* has a property *PROPERTY* with an incorrect value *INCORRECT_VALUE*. The operation cannot proceed until the property has the correct value *CORRECT_VALUE*. Take administrative action to change the value and try again.

Explanation

The specified object has the specified property of the specified type with the specified incorrect value. The property most likely reflects the state of the object.

Action

Change the state to the required value, and resubmit the command.

CMMVC6133E Required *TYPE* property *PROPERTY* not found

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6134E No argument for *OPTION*

Explanation

No argument has been supplied for the specified option, which requires an argument.

Action

Supply an argument, and resubmit the command.

CMMVC6135E Argument *VALUE* for *OPTION* is not valid.

Explanation

The specified argument that you have supplied is not valid for the specified option.

Action

Supply an valid argument, and resubmit the command.

CMMVC6136W No SSH key file *FILENAME*

Explanation

The specified file, which should contain the SSH key, is not present and will not be restored. The backup operation will continue.

Action

No action is required. You might have to manually restore the key.

CMMVC6137W No SSH key file *FILENAME*; key not restored

Explanation

An SSH key cannot be restored because the specified file, which is expected to contain the SSH key, is not present. The restore operation will continue.

Action

After the restore is complete, locate the file containing the key, and perform one of the following actions: (1) Rename the file so that it has the correct name, and resubmit the command. (2) Restore the key manually using the svctask addsshkey command.

CMMVC6138E *OPTION* is required

Explanation

An option is missing. The option might be listed as optional, but circumstances make the option mandatory.

Action

Supply the option, and resubmit the command.

CMMVC6139E Incorrect XML tag nesting in *FILENAME*

Explanation

There is a problem with the content of a configuration file. There is a problem parsing the XML in the file, because the XML records are not consistent. The file might be corrupt, or the file has been truncated.

Action

Replace this copy with a good copy, and resubmit the command. If the problem persists, contact IBM technical support for assistance.

CMMVC6140E No default name for type *TYPE*

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6141E The option *OPTION* does not support an argument.

Explanation

An argument has been supplied for an option that does not support one.

Action

Remove the argument, and resubmit the command.

CMMVC6142E Existing *OBJECT_TYPE OBJECT_NAME* has a non-default name.

Explanation

The specified object in the target default cluster has a non-default name. This suggests that the cluster was customized. The cluster is therefore not suitable for restoration.

Action

Reset the cluster as per the instructions for restoring the cluster configuration, and resubmit the command.

CMMVC6143E The required configuration file *FILENAME* does not exist.

Explanation

A file that is critical for successful operation is missing.

Action

Not applicable.

CMMVC6144W The object with default name *NAME* has been restored as *SUBSTITUTE_NAME*.

Explanation

An object with a default name bas been restored with a different name. Ensure that you account for this name change when using the restored cluster in the future. To avoid this problem in the future, choose an appropriate name for each object in the cluster.

Action

Choose an appropriate name for each object in the cluster.

CMMVC6145I First use the *COMMAND* -prepare command.

Explanation

This advisory is given prior to CMMVC6103E when an intermediate file is missing.

Action

Not applicable.

CMMVC6146E Problem parsing *OBJECT_TYPE* data: *LINE*

Explanation

An unexpected error has occurred.

Action

Contact the support center.

CMMVC6147E *TYPE NAME* has a name beginning with *PREFIX*.

Explanation

An object has been encountered that has a name beginning with the specified reserved prefix. The only valid reason for an object with this kind of name is that a restoration command did not complete successfully.

Action

Ensure that no object uses the reserved prefix in its name, and resubmit the command.

CMMVC6148E Target cluster has *NUMBER_OF_EXISTING_OBJECTS* objects of type *TYPE* instead of *NUMBER_OF_REQUIRED_OBJECTS*.

Explanation

The target cluster does not have the specified required number of objects of the specified type.

Action

Correct the problem, and resubmit the command.

CMMVC6149E An action is required.

Explanation

An action is required to run the command.

Action

Supply an action, and resubmit the command.

CMMVC6150E The action *ACTION* is not valid.

Explanation

The specified action that you have entered is not valid.

Action

Specify a valid action, and resubmit the command.

CMMVC6151E The option *OPTION* is not valid.

Explanation

The specified option that you have entered is not valid.

Action

Specify a valid option, and resubmit the command.

CMMVC6152E VDisk *VDISK_NAME* instance number *INSTANCE_NUMBER* is not valid.

Explanation

The VDisk cannot be restored because the instance number, which must be a hexadecimal number, is not valid.

Action

Contact IBM technical support for assistance.

CMMVC6153E *OBJECT* not consistent with *ACTION*

Explanation

The specified object is not supported for the specified action.

Action

Remove the object, and resubmit the command.

CMMVC6154E Required *OBJECT_TYPE* property *PROPERTY_NAME* has a null value.

Explanation

An unexpected error has occurred.

Action

Contact IBM technical support for assistance.

CMMVC6155I The command *COMMAND* processing has completed successfully.

Explanation

Only information and warning messages are issued.

Action

Not applicable.

CMMVC6156W *COMMAND* processing completed with errors.

Explanation

Processing was not successful.

Action

Not applicable.

CMMVC6164E The SVCCONFIG CRON job, which runs overnight on a daily overnight, has failed.

Explanation

The SVCCONFIG CRON job, which runs overnight on a daily overnight, has failed.

Action

Resolve any hardware and configuration problems that you are experiencing on the 2145 cluster. If the problem persists, contact IBM technical support for assistance.

CMMVC6165E The target is not the original configuration node with a WWNN of *WWNN_VALUE*.

Explanation

A backup configuration can only be restored to the original configuration node.

Action

Recreate the default cluster with the correct configuration node, and resubmit the command.

CMMVC6166E The property *PROPERTY* of the object *OBJECT* has changed during svcconfig restore -execute.

Explanation

The integrity of the restoration cannot be guaranteed.

Action

Resubmit the command from svcconfig restore -prepare.

CMMVC6181E The target cluster contains an object that has a counterpart in the configuration to be restored, and has the correct ID.

Explanation

The indicated property has an unexpected value.

Action

Check that the correct (matching) backup configuration file (svc.config.backup.xml) is being provided and if it is, use the -force option to ignore the discrepancy. Otherwise, provide the correct file and try again.

CMMVC6182W An object that does not contribute to the fabric of the configuration cannot be restored because its configuration does not permit it to be created.

Explanation

An object that does not contribute to the fabric of the configuration cannot be restored because its configuration does not permit it to be created. For example, a host can only be created if it has at least one port.

Action

Not applicable.

CMMVC6186W The IO group *IO_GROUP_NAME* has been restored with ID *ID_VALUE* instead of *ID_VALUE*.

Explanation

This can occur when the configuration node is different from the node that was used to create the original cluster. This affects the SCSI Inquiry value for the IO group.

Action

Not applicable.

CMMVC6200E The action failed because of incompatible software.

Explanation

The software version on one or more nodes is incompatible with the new version.

Action

Refer to the compatibility requirements for the software version you are adding. Update the cluster to meet the compatibility requirements, and then perform the upgrade.

CMMVC6201E The node could not be added because of incompatible software. The status code is *STATUS_CODE*.

Explanation

The node could not be added because of incompatible software.

Action

Upgrade the software on the node that has been rejected to the same level of software as the cluster to which it will be added, and resubmit the command.

CMMVC6202E The cluster was not modified because the IP address is not valid.

Explanation

An attempt was made to change the IP address of a cluster to an invalid address.

Action

Correct the address and re-issue the command.

CMMVC6203E The action failed because the directory that was specified was not one of the following directories: /dumps, /dumps/iostats, /dumps/iotrace, /dumps/feature, /dumps/config, /dumps/elogs, /dumps/ec or /dumps/pl.

Explanation

An attempt was made to clear a file from, or copy a file to, an invalid directory.

Action

Ensure that the command accesses a valid directory.

CMMVC6204E The action failed as the resulting disk size would be less than, or equal to, zero.

Explanation

An attempt was made to shrink a disk, however the resulting size would have been less than or equal to zero.

Action

Not applicable

CMMVC6205E Metadata recovery can not use the provided mdisk id - invalid or destroyed.

Explanation

Metadata recovery can not use the provided mdisk id - invalid or destroyed.

CMMVC6206E The software upgrade failed as a file containing the software for the specified MCP version was not found.

Explanation

There are two files required to successfully complete a software upgrade. One file contains the files that make up the base operating system, and the other file contains the 2145 software. This message appears if the OS version is incompatible with the 2145 software.

Action

Upload two compatible files, and resubmit the command.

CMMVC6207E The action failed because the virtual disk (VDisk) is part of a Remote Copy mapping.

Explanation

An action was performed against a VDisk that is part of a Remote Copy mapping.

Action

Remove the VDisk from the Remote Copy mapping before resubmitting the command.

CMMVC6208E The action failed because the virtual disk (VDisk) is part of a FlashCopy mapping.

Explanation

An action was performed against a VDisk that is part of a FlashCopy mapping.

Action

Remove the VDisk from the FlashCopy mapping before re-issuing the command.

CMMVC6209E The FlashCopy mapping or consistency group could not be started in a reasonable time. The mapping or group is instead being prepared.

Explanation

The FlashCopy mapping or consistency group could not be started in a reasonable time. The mapping or group is instead being prepared.

Action

Resubmit the command.

CMMVC6210E The command has failed because a virtual medium error exists on the image mode VDisk or copy.

Explanation

When you submit this command, you cannot specify an image mode VDisk that has a virtual medium error on the VDisk or on any copy of the VDisk because the medium errors cannot be maintained on the ejected MDisk image copy.

Action

If an exact image copy is required, ensure that there is no virtual medium error on the image mode VDisk that you specify or on any of its copies, and resubmit the command.

If an exact copy is not required, you can use the -force option of the command, but all of the virtual medium errors will be lost.

CMMVC6211E The command failed as a migrate to image was in progress.

Explanation

An attempt was made to execute a command against a VDisk that was involved in a migrate to image operation.

Action

Wait for the migration to complete and re-issue the command.

CMMVC6212E The command failed because data in the cache has not been committed to disk.

Explanation

The command failed because data in the cache has not been committed to disk.

CMMVC6213E You are trying to recover region data that was created by a code level different from the one you are currently running on the node.

Explanation

You are trying to recover region data that was created by a code level different from the one you are currently running on the node.

CMMVC6214E Failed to recreate the cluster you are trying to rebuild.

Explanation

Failed to recreate the cluster you are trying to rebuild.

CMMVC6215E The FlashCopy mapping was not created or modified because the consistency group already contains the maximum number of mappings.

Explanation

An attempt was made to create a FlashCopy mapping in, or move a FlashCopy mapping to, a consistency group that has the maximum number of FlashCopy mappings that it can contain.

Action

Create or move the FlashCopy mapping in another consistency group or remove an existing FlashCopy mapping from the desired group and then re-issue the command.

CMMVC6216E The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is a member of a Remote Copy mapping.

Explanation

The Remote Copy relationship was not created because the master or auxiliary virtual disk (VDisk) is a member of a Remote Copy mapping.

Action

Select a different VDisk to make up the mapping.

CMMVC6217E The maximum number of hosts for the cluster is already configured.

Explanation

You must remove at least one host definition before you can resubmit the command.

Action

Determine whether the action is required.

If the action is required, review the current configuration to determine whether any current host definitions are not required. Remove at least one host definition that is not required, and resubmit the command.

CMMVC6218E The maximum number of host/IO group pairs for the cluster is already configured.

Explanation

You must remove at least one host/IO group pair definition before you can resubmit the command.

Action

Determine whether the action is required.

If the action is required, review the current configuration to determine whether any current host/IO group pair definitions are not required. Remove at least one host/IO group pair definition that is not required, and resubmit the command.

CMMVC6219E The maximum number of WWPNs for the cluster is already configured.

Explanation

You must remove at least one WWPN definition before you can resubmit the command.

Action

Determine whether the action is required.

If the action is required, review the current configuration to determine whether any current WWPN definitions are not required. Remove at least one WWPN definition that is not required, and resubmit the command.

CMMVC6220E The maximum number of hosts for one or more IO groups is already configured.

Explanation

You must remove at least one host/IO group pair definition from the I/O group that you have specified before you can resubmit the command.

Action

Determine whether the action is required.

If the action is required, review the current configuration to determine whether any current host/IO group pair definitions for the I/O group that you have specified are not required. Remove at least one host/IO group pair definition that is not required from the I/O group that you have specified, and resubmit the command.

CMMVC6221E The maximum number of WWPNs for one or more IO groups is already configured.

Explanation

You must remove at least one WWPN definition from the I/O group that you have specified before you can resubmit the command.

Action

Determine whether the action is required.

If the action is required, review the current configuration to determine whether any current WWPN definitions for the I/O group that you have specified are not required. Remove at least one WWPN definition that is not required from the I/O group that you have specified, and resubmit the command.

CMMVC6222E The maximum number of WWPNs for the host is already configured.

Explanation

You must remove at least one WWPN definition for the host that you have specified before you can resubmit the command.

Action

Determine whether the action is required.

If the action is required, review the current configuration to determine whether any current WWPN definitions for the host that you have specified are not required. Remove at least one WWPN definition that is not required for the host that you have specified, and resubmit the command.

CMMVC6223E The host does not belong to one or more of the IO groups specified or inferred.

Explanation

The host does not belong to one or more of the IO groups specified or inferred.

Action

Specify a host/IO group combination that is currently defined, and resubmit the command.

CMMVC6224E The host already belongs to one or more of the IO groups specified.

Explanation

The host already belongs to one or more of the IO groups specified.

Action

None.

CMMVC6225E An IO group cannot be removed from a host because of one or more associated vdisks.

Explanation

An IO group cannot be removed from a host because of one or more associated vdisks.

CMMVC6226E The action was not completed because the cluster has reached the maximum number of extents in MDisk Groups.

Explanation

The cluster has reached the maximum number of extents in the MDisk Group; therefore, the action did not complete. You are attempting to use additional extents, for example by creating or expanding a VDisk. The action cannot be initiated because it would cause the maximum number of extents for a cluster to be exceeded.

Action

Free up extents by deleting other VDisks, and resubmit the command.

CMMVC6227I The package installed successfully.

Explanation

The package installed successfully.

Action

None.

CMMVC6228E The cluster was recovered and the CLI functionality is limited until the cause of the failure is determined and any corrective action taken. Contact IBM technical support for assistance.

Explanation

The cluster was recovered and the CLI functionality is limited.

Action

Contact IBM technical support for assistance.

CMMVC6229E The action failed as the SSH key has been revoked.

Explanation

The action failed as the SSH key has been revoked.

CMMVC6230E The action failed as the SSH key index (SSH_LABEL_ID) is invalid.

Explanation

The action failed as the SSH key index (SSH_LABEL_ID) is invalid.

CMMVC6231E The action failed as the audit table is full.

Explanation

The action failed as the audit table is full.

Action

Save the audit log to disk, and resubmit the command.

CMMVC6232E This operation cannot be performed because the cluster is currently aborting the previous software upgrade command.

Explanation

This operation cannot be performed because the cluster is currently aborting the previous software upgrade command.

Action

Wait until the previous software upgrade command has aborted successfully, and resubmit the command.

CMMVC6233E This operation cannot be performed because, either a software upgrade has not been started, or a software upgrade is in progress but is not in a state where it can be aborted.

Explanation

This operation cannot be performed because the software upgrade is making progress.

CMMVC6234E The upgrade cannot be aborted because at least one node has already committed to a new code level.

Explanation

The upgrade cannot be aborted because at least one node has already committed to a new code level.

CMMVC6235E An invalid response has been entered. The command has not been executed. Input is case sensitive. Enter either yes or no.

Explanation

An invalid response has been entered. The command has not been executed. Enter either yes or no.

Action

Enter either yes or no.

CMMVC6236E The command has not completed. A limited availability parameter has been entered without the required environment setting being set.

Explanation

The command has not completed. A limited availability parameter has been entered without the required environment setting being set.

CMMVC6237E The command failed as the remote cluster does not support global mirror.

Explanation

The command failed as the remote cluster does not support global mirror.

CMMVC6238E The copy type differs from other copies already in the consistency group.

Explanation

The copy type differs from other copies already in the consistency group.

Action

Ensure that the copy type of the mapping that you are attempting to add is the same copy type as the mappings in the consistency group to which you are attempting to add the mapping, and resubmit the command.

CMMVC6239E The FlashCopy mapping was not prepared because the mapping or consistency group is in the stopping state. The mapping or consistency group must first complete the stop operation and then be prepared

Explanation

You cannot prepare a FlashCopy mapping or consistency group when the FlashCopy mapping or consistency group is in the stopping state. If you want to prepare a FlashCopy mapping or consistency group, the FlashCopy mapping or consistency group must be in the Stopped or idle_or_copied state.

Action

Wait until the FlashCopy mapping or consistency group reaches the Stopped or idle_or_copied state and then resubmit the command.

CMMVC6240E The properties of the FlashCopy mapping were not modified because the mapping or consistency group is in the stopping state.

Explanation

You cannot modify the consistency group of a FlashCopy mapping when the FlashCopy mapping is in the stopping state. If you want to modify the consistency group of a FlashCopy mapping, the FlashCopy mapping must be in the Stopped or idle_or_copied state.

Action

Wait until the FlashCopy mapping reaches the Stopped or idle_or_copied state and then resubmit the command.

CMMVC6241E The FlashCopy mapping was not deleted because the mapping or consistency group is in the stopping state. The mapping or consistency group must be stopped first.

Explanation

You cannot delete a FlashCopy mapping or consistency group when the FlashCopy mapping or consistency group is in the stopping state. If you want to delete a FlashCopy mapping or consistency group, the FlashCopy mapping or consistency group must be in the Stopped or idle_or_copied state.

Action

Wait until the FlashCopy mapping or consistency group reaches the Stopped or idle_or_copied state and then resubmit the command.

CMMVC6242E The FlashCopy mapping or consistency group was not started because the mapping or consistency group is in the stopping state. The mapping or consistency group must first complete the stop operation and then be prepared.

Explanation

You cannot start a FlashCopy mapping or consistency group when the FlashCopy mapping or consistency group is in the stopping state. If you want to start a FlashCopy mapping or consistency group, the FlashCopy mapping or consistency group must be in the Prepared state.

Action

Wait until the FlashCopy mapping or consistency group reaches the Stopped or idle_or_copied state and then prepare the FlashCopy mapping or consistency group before starting it.

CMMVC6243E The FlashCopy mapping or consistency group was not stopped because the mapping or consistency group is already in the stopping state.

Explanation

A Stop FlashCopy mapping or consistency group task has already been submitted and is still in progress. When the task has completed successfully, the FlashCopy mapping or consistency group state will change to Stopped.

Action

None.

CMMVC6244E The FlashCopy mapping was not created because the source virtual disk (VDisk) cannot be the target for a FlashCopy mapping.

Explanation

A VDisk cannot simultaneously be both the source of a FlashCopy mapping and the target of a FlashCopy mapping. The source VDisk that you have specified is currently defined as the target of a FlashCopy mapping.

Action

You have two options. One option is specify a different source VDisk and resubmit the command. The other option is delete the existing FlashCopy mapping that defines the source VDisk that you have specified as the target VDisk, and resubmit the command.

CMMVC6245E The FlashCopy mapping was not created because the source virtual disk (VDisk) is already in the maximum number of FlashCopy mappings.

Explanation

The number of FlashCopy mappings in which a Vdisk can be defined as the source Vdisk is limited. The source Vdisk that you have specified cannot be defined to another FlashCopy mapping because it is already defined as the source Vdisk to the maximum number of FlashCopy mappings.

Action

You have two options. One option is specify a different source VDisk and resubmit the command. The other option is delete one of the existing FlashCopy mappings that contains the source VDisk and resubmit the command.

CMMVC6246E The FlashCopy mapping was not created because the target virtual disk (VDisk) is already a source vdisk in a FlashCopy mapping.

Explanation

A VDisk cannot simultaneously be both the source of a FlashCopy mapping and the target of a FlashCopy mapping. The target VDisk that you have specified is currently defined as the source of a FlashCopy mapping.

Action

You have two options. One option is specify a different target VDisk and resubmit the command. The other option is delete all of the existing FlashCopy mappings that contain the target VDisk that you have specified and resubmit the command.

CMMVC6247E The FlashCopy mapping was not created because the target virtual disk (VDisk) is already a target vdisk in a FlashCopy mapping.

Explanation

A VDisk cannot simultaneously be the target of more than one FlashCopy mapping. The target VDisk that you have specified is currently defined as the target of another FlashCopy mapping.

Action

You have two options. One option is specify a different target VDisk and resubmit the command. The other option is delete the existing FlashCopy mapping that contains the target VDisk that you have specified and resubmit the command.

CMMVC6248E The command failed because the authorization table is full.

Explanation

The command failed because the authorization table is full.

CMMVC6249E The command failed because the authorization record was not found or is already set to the default role.

Explanation

The command failed because the authorization record was not found or is already set to the default role.

CMMVC6250E The command failed because the authorization record is not set to the default role. Use rmauth to set the default role.

Explanation

The command failed because the authorization record is not set to the default role.

Action

Use rmauth to set the default role.

CMMVC6251E The command failed because the specified role was not found.

Explanation

The command failed because the specified role was not found.

CMMVC6252E The command failed authorization because the session ssh key is invalid or was deleted.

Explanation

The command failed authorization because the session ssh key is invalid or was deleted.

CMMVC6253E The command failed authorization because the session ssh key does not have the requisite role.

Explanation

The command failed authorization because the session ssh key does not have the requisite role.

CMMVC6254E The command failed because the specified ssh key was not found. NOTE This command must specify an admin key.

Explanation

The command failed because the specified ssh key was not found. NOTE This command must specify an admin key.

CMMVC6255E The command cannot set the authorization record to the default role. Use rmauth to set the default role.

Explanation

The command can not set the authorization record to the default role.

Action

Use rmauth to set the default role.

CMMVC6263E The command failed because the ssh key already exists or there is a duplicate ssh key.

Explanation

You have attempted to add an ssh key that already exists, and may have a different authorization level associated with it.

Action

Add a different ssh key if the existing ssh key of the same type does not have the authority level that you require.

CMMVC6269E Sendmail error EX_USAGE. A command or configuration line has been used incorrectly.

Explanation

The send email task has failed because a command or a configuration line has been used incorrectly.

Action

Ensure that the email settings are correct, and resubmit the task.

CMMVC6270E Sendmail error EX_DATAERR. Address is wrong, or the message is too large for the mailbox.

Explanation

The send email task has failed because the message sent is too large or a recipient address is incorrect.

Action

Ensure that all addresses are correct and that the message is not too large, and resubmit the task.

CMMVC6271E Sendmail error EX_NOINPUT. An input file (not a system file) did not exist or was not readable.

Explanation

The send email task has failed because a file is missing or cannot be read.

Action

Ensure that the email system is configured correctly. Ensure that access permissions have been specified correctly for all email configuration files, and resubmit the task.

CMMVC6272E Sendmail error EX_NOUSER. The sendmail command could not recognize a specified user ID.

Explanation

The send email task has failed because the User ID that you specified does not exist.

Action

Specify a defined User ID, and resubmit the task.

CMMVC6273E Sendmail error EX_NOHOST. The sendmail command could not recognize the specified host name.

Explanation

The send email task has failed because the host is not known to the email system.

Action

Ensure that you have configured the SMTP environment correctly. Ensure that you specify a defined host, and resubmit the task.

CMMVC6274E Sendmail error EX_UNAVAILABLE. A required system resource is not available.

Explanation

The send email task has failed because a required system resource is not available.

Action

Ensure that you have configured the SMTP environment correctly, and resubmit the task.

CMMVC6275E Sendmail error EX_SOFTWARE. An internal software error occurred (including bad arguments).

Explanation

The send email task has failed because an incorrect parameter or parameter value has been detected.

Action

Ensure that you have configured the SMTP environment correctly. Specify only supported parameters and parameter values, and resubmit the task.

CMMVC6276E Sendmail error EX_OSERR. A system resource error prevented the sending of an email.

Explanation

The send email task has failed because a system resource error has occurred.

Action

Ensure that you have configured the SMTP environment correctly, and resubmit the task.

CMMVC6277E Sendmail error EX_OSFILE. Failed to open a critical system file.

Explanation

The send email task has failed because a required system file cannot be opened.

Action

Ensure that the email system is configured correctly. Ensure that access permissions have been specified correctly for all email configuration files, and resubmit the task.

CMMVC6278E Sendmail error EX_CANTCREAT. An output file could not be written to by sendmail.

Explanation

The send email task has failed because the system cannot write to a required output file.

Action

Ensure that the email system is configured correctly. Ensure that access permissions have been specified correctly for all email configuration files, and resubmit the task.

CMMVC6279E Sendmail error EX_IOERR. A system I/O error occurred during a sendmail operation. This could be due to a disk failure.

Explanation

The send email task has failed because a write or read I/O operation has failed. This error might be caused by a disk device failure.

Action

Correct the root cause of the I/O failure, and resubmit the task.

CMMVC6280E Sendmail error EX_TEMPFAIL. The sendmail command could not create a connection to a remote system.

Explanation

The send email task has failed because the sendmail application cannot establish a connection to the remote system.

Action

Ensure that the network connection to the remote system is functioning correctly, and resubmit the task.

CMMVC6281E Sendmail error EX_PROTOCOL. The remote system returned something that was incorrect during a protocol exchange.

Explanation

The send email task has failed because an error in the protocol exchange has occurred.

Action

Ensure that the email system is configured correctly. Ensure that you have configured the SMTP environment correctly, and resubmit the task.

CMMVC6282E Sendmail error EX_NOPERM. The user does not have permission to perform the requested operation.

Explanation

The send email task has failed because the User ID does not have authorization to submit the task.

Action

Ensure that authorizations for your User ID in the email and SMTP configurations are correct, and resubmit the task.

CMMVC6283E Sendmail error EX_CONFIG. There is a fatal problem with the sendmail configuration.

Explanation

The send email task has failed because the sendmail configuration is not correct.

Action

Ensure that the email system is configured correctly. Ensure that you have configured the SMTP environment correctly, and resubmit the task.

CMMVC6284E An unknown error occurred. Please ensure your SMTP server is running.

Explanation

The send email task has failed because an unexpected error has occurred.

Action

Ensure that the SMTP server is running, and resubmit the task.

CMMVC6285E The email command timed out. Please check your email server settings as listed on the SVC.

Explanation

The send email task has failed because a command timeout has occurred.

Action

Ensure that your system settings match those recommended in the sendmail application documentation, and resubmit the task.

CMMVC6286E The email service has not been enabled.

Explanation

The send email task has failed because the email application is not enabled.

Action

Enable the email application, and resubmit the task.

CMMVC6287E The user specified does not exist.

Explanation

You must specify a User ID that exists.

Action

Ensure that the User ID that you specify is defined, and resubmit the task.

CMMVC6288E The FlashCopy mapping or consistency group could not be started because a source vdisk is the target of another FC Map that is keeping the vdisk inaccessible.

Explanation

You cannot start a FlashCopy mapping or consistency group when a source vdisk in the FlashCopy mapping or consistency group is the target vdisk of another FlashCopy mapping that is holding the vdisk as inaccessible. The task cannot be initiated because a source vdisk in the FlashCopy mapping or consistency group that you are attempting to start is the target of another FlashCopy mapping that is either prepared, preparing, stopped or stopping with a progress of less than 100%.

Action

Ensure that you have selected to start the correct FlashCopy mapping or consistency group. Ensure that none of the source vdisks in the FlashCopy mapping or consistency group that you specify are target vdisks in another FlashCopy mapping that is prepared, preparing, stopped or stopping with a progress of less than 100%. Resubmit the task.

CMMVC6289E The command failed because the virtual disk (VDisk) is pending synchronization.

Explanation

This error occurs when at least one of the virtual disk copies is offline.

Action

Fix all of the errors that are associated with the virtual disk copies, and resubmit the command.

CMMVC6290E The command failed because the virtual disk (VDisk) has image mode copies, is pending synchronization and -force has not been specified.

Explanation

This error occurs when at least one of the virtual disk copies is offline.

Action

Perform one of the following actions:

- Fix all of the errors that are associated with the virtual disk copies, and resubmit the command.
- Resubmit the command and specify the -force parameter.

NOTE: When you specify the -force parameter with the command that caused this error, the image mode virtual disk copy is no longer guaranteed to have the correct virtual disk data.

CMMVC6291E The command failed because the virtual disk (VDisk) is pending synchronization and -force has not been specified.

Explanation

The command failed because the virtual disk (VDisk) is pending synchronization and -force has not been specified.

Action

Perform one of the following actions:

- Fix all of the errors that are associated with the virtual disk copies, and resubmit the command.
- Resubmit the command and specify the -force parameter.

NOTE: When you specify the -force parameter with the command that caused this error, the entire virtual disk copy is resynchronized.

CMMVC6292E The command failed because a repair action is in progress for the virtual disk (VDisk).

Explanation

You cannot submit this command while a repair action is in progress for the virtual disk (VDisk).

Action

Use the svcinfo lsrepairvdiskcopyprogress command to view the repair progress. Wait for the virtual disk repair process to complete. If you want the repair process to complete more quickly, increase the rate by submitting an svctask chvdisk command. Once the repair action has completed, resubmit the command that caused this error.

CMMVC6296E One or more managed disks (MDisks) have failed validation tests. The first failing MDisk ID is *MDISK_ID*.

Explanation

When you add a managed MDisk to an MDisk group, the new MDisk is validated to ensure that adding it to the MDisk group will not adversely impact the MDisk group status. Either the current status of the MDisk has not allowed the validation to be performed, or the validation has failed. Note: You cannot add Read Only or faulty MDisks to an MDisk group.

Action

- If the MDisk identity has changed since it was last discovered, submit the command-line interface command svctask detectmdisk, which might correct the problem.
- Check switch zoning and logical unit presentation on the controller to ensure that the MDisk is physically and logically connected to all of the nodes in this cluster.

- Ensure that the controller settings are correct and that the MDisk logical unit is correctly configured.
- Ensure that the MDisk logical unit state is one that passes the validation. A Read Only or faulty MDisk fails the validation.
- View the cluster error log for more information about the failed validation.

CMMVC6297E One or more managed disks (MDisks) have timed out during validation tests. The first failing MDisk ID is *MDISK_ID*.

Explanation

When you add a managed MDisk to an MDisk group, the new MDisk is validated to ensure that adding it to the MDisk group will not adversely impact the MDisk group status. The current status of the MDisk permits the validation to be initiated, but the allotted time for the validation process elapsed before the validation process had completed. Note: You cannot add Read Only or faulty MDisks to an MDisk group.

Action

- Ensure that the controller settings are correct and that the MDisk logical unit is correctly configured.
- Ensure that the MDisk logical unit state is one that passes the validation. A Read Only or faulty MDisk fails the validation.
- Check the fibre-channel fabric and storage controller for faults that might reduce the reliability of cluster communication with the MDisk.
- View the cluster error log for more information about the failed validation.

CMMVC6298E The command failed because a target VDisk has dependent FlashCopy mappings.

Explanation

The target VDisk of the FlashCopy mapping, or the target VDisk of at least one of the FlashCopy mappings in the consistency group, has other FlashCopy mappings that are dependent on the data on the target VDisk.

Action

Use the lsvdiskdependentmaps command and specify the target VDisk to determine which FlashCopy mappings are dependent on the target VDisk. Either wait for these mappings to reach the idle_or_copied state, or stop these mappings. Resubmit the command that produced this error.

CMMVC6299E The create failed because the source and target VDisks are members of FlashCopy mappings that have different grain sizes.

Explanation

All FlashCopy mappings that are in a tree of connected mappings must have the same grain size. The new FlashCopy mapping that you attempted to create would have linked two existing trees that have different grain sizes.

Action

You have three options. The first option is to resubmit the command and specify a different source or target VDisk. The second option is to delete all of the existing mappings that contain the source VDisk and resubmit the command. The third option is to delete all of the existing mappings that contain the target VDisk and resubmit the command.

CMMVC6300E The create failed because the source and target VDisks are members of FlashCopy mappings that belong to different I/O groups.

Explanation

All FlashCopy mappings in a tree of connected mappings must be in the same I/O group. The new FlashCopy mapping that you attempted to create would have linked two existing trees that are in different I/O groups.

Action

You have three options. The first option is to resubmit the command and specify a different source or target VDisk. The second option is to delete all of the existing mappings that contain the source VDisk and resubmit the command. The third option is to delete all of the existing mappings that contain the target VDisk and resubmit the command.

CMMVC6301E The create failed because the specified consistency group does not exist.

Explanation

The FlashCopy mapping was not created because the consistency group that you specified does not exist. You must create a consistency group before you can place a mapping in that group.

Action

Either create the FlashCopy consistency group that you specified and resubmit the command, or resubmit the command and specify an existing consistency group.

CMMVC6302E The create failed because the resulting tree of FlashCopy mappings would exceed the upper limit.

Explanation

Either the source VDisk or the target VDisk, or both, are already members of other FlashCopy mappings. The FlashCopy mapping was not created because the new FlashCopy mapping that you attempted to create would have linked two existing mapping trees into a single tree that exceeds the maximum number of mappings that are supported for a single tree.

Action

You have two options. The first option is to resubmit the command and specify a different source or target VDisk. The second option is to delete a sufficient number of the existing FlashCopy mappings in which either the source or the target VDisk is a member so that the combined mapping tree does not exceed the maximum number of mappings that are supported for a single tree, and resubmit the command.

CMMVC6303E The create failed because the source and target VDisks are the same.

Explanation

A particular Vdisk cannot be both the source and the target in a FlashCopy mapping. The FlashCopy mapping was not created because you have specified the same VDisk as both the source and the target.

Action

Resubmit the command and specify source and target VDisks that are not identical.

CMMVC6304E The create failed because the source VDisk does not exist.

Explanation

You must specify an existing VDisk as the source of a FlashCopy mapping. The FlashCopy mapping was not created because the source VDisk that you specified does not exist.

Action

Either create the source VDisk that you specified and resubmit the command, or resubmit the command and specify an existing VDisk as the source.

CMMVC6305E The create failed because the target VDisk does not exist.

Explanation

You must specify an existing VDisk as the target of a FlashCopy mapping. The FlashCopy mapping was not created because the target VDisk that you specified does not exist.

Action

Either create the target VDisk that you specified and resubmit the command, or resubmit the command and specify an existing VDisk as the target.

CMMVC6306E The create failed because the source VDisk is the member of a FlashCopy mapping whose grain size is different to that specified.

Explanation

All FlashCopy mappings that are in a tree of connected mappings must have the same grain size. The FlashCopy mapping was not created because the source VDisk that you specified is either the source or the target VDisk of another FlashCopy mapping, and the grain size of the other mapping is different from the grain size that you specified for the mapping that you attempted to create.

Action

You have two options. The first option is to delete all of the FlashCopy mappings that contain the source VDisk that you specified where the grain size of the FlashCopy mapping is different from the grain size that you specified, and resubmit the command. The second option is to resubmit the command and do not specify the grain size attribute.

CMMVC6307E The create failed because the target VDisk is the member of a FlashCopy mapping whose grain size is different to that specified.

Explanation

All FlashCopy mappings that are in a tree of connected mappings must have the same grain size. The FlashCopy mapping was not created because the target VDisk that you specified is either the source or the target VDisk of another FlashCopy mapping, and the grain size of the other mapping is different from the grain size that you specified for the mapping that you attempted to create.

Action

You have two options. The first option is to delete all of the FlashCopy mappings that contain the target VDisk that you specified where the grain size of the FlashCopy mapping is different from the grain size that you specified, and resubmit the command. The second option is to resubmit the command and do not specify the grain size attribute.

CMMVC6308E The create failed because the source VDisk is the member of a FlashCopy mapping whose IO group is different to that specified.

Explanation

All FlashCopy mappings in a tree of connected mappings must be in the same I/O group. The FlashCopy mapping was not created because the source VDisk that you specified is the source or target VDisk in another FlashCopy mapping and the I/O group of the other FlashCopy mapping is different from the I/O group that you specified.

Action

You have two options. The first option is to delete all of the FlashCopy mappings that contain the source VDisk that you specified where the FlashCopy mapping is in a different I/O group from the I/O group that you specified, and resubmit the command. The second option is to resubmit the command and do not specify the I/O group attribute. If you perform the second option, the default value of the I/O group attribute is used.

CMMVC6309E The create failed because the target VDisk is the member of a FlashCopy mapping whose IO group is different to that specified.

Explanation

All FlashCopy mappings in a tree of connected mappings must be in the same I/O group. The FlashCopy mapping was not created because the target VDisk that you specified is the source or target VDisk in another FlashCopy mapping and the I/O group of the other FlashCopy mapping is different from the I/O group that you specified.

Action

You have two options. The first option is to delete all of the FlashCopy mappings that contain the target VDisk that you specified where the FlashCopy mapping is in a different I/O group from the I/O group that you specified, and resubmit the command. The second option is to resubmit the command and do not specify the I/O group attribute. If you perform the second option, the default value of the I/O group attribute is used.

CMMVC6310E The modify failed because the specified FlashCopy mapping does not exist.

Explanation

You cannot modify a FlashCopy mapping that does not exist. The modify command failed because the FlashCopy mapping that you specified does not exist.

Action

Resubmit the command and specify an existing FlashCopy mapping.

CMMVC6311E The command failed because the source VDisk is the target of a FlashCopy mapping that is in the specified consistency group.

Explanation

A particular VDisk cannot be both the source of one FlashCopy mapping and the target of another FlashCopy mapping in the same consistency group. The FlashCopy mapping was not created because the source VDisk of the FlashCopy mapping that you attempted to create is already the target VDisk of a FlashCopy mapping in the consistency group that you specified.

Action

Resubmit the command and specify a different consistency group.

CMMVC6312E The command failed because the target VDisk is the source of a FlashCopy mapping that is in the specified consistency group.

Explanation

A particular VDisk cannot be both the source of one FlashCopy mapping and the target of another FlashCopy mapping in the same consistency group. The FlashCopy mapping was not created because the target VDisk of the FlashCopy mapping that you attempted to create is already the source VDisk of a FlashCopy mapping in the consistency group that you specified.

Action

Resubmit the command and specify a different consistency group.

CMMVC6313E The command failed because the specified background copy rate is invalid.

Explanation

The command failed because the background copy rate that you specified is not a supported value.

Action

Either resubmit the command and specify a supported value for the background copy rate, or resubmit the command and do not specify the background copy rate attribute. If you do not specify the background copy rate attribute, the default background copy rate value is used.

CMMVC6314E The command failed because the specified cleaning rate is not valid.

Explanation

The command failed because the cleaning rate that you specified is not a supported value.

Action

Either resubmit the command and specify a supported value for the cleaning rate, or resubmit the command and do not specify the cleaning rate attribute. If you do not specify the cleaning rate attribute, the default cleaning rate value is used.

CMMVC6315E The command failed because the specified grain size is not valid.

Explanation

The command failed because the grain size that you specified is not a supported value.

Action

Either resubmit the command and specify a supported value for the grain size, or resubmit the command and do not specify the grain size attribute. If you do not specify the grain size attribute, the default grain size value is used.

CMMVC6319E The command failed as a combination of IPV4 and IPV6 parameters were entered.

Explanation

The task accepts either IPV4 or IPV6 parameters. You cannot specify a combination of IPV4 and IPV6 parameters for this task.

Action

Specify only IPV4 or only IPV6 parameters, and resubmit the task.

CMMVC6320E The command failed as the IPV4 address supplied was invalid.

Explanation

The value that you entered for an IPV4 address is not a valid IPV4 address.

Action

Specify a valid IPV4 address, and resubmit the task.

CMMVC6321E The command failed as the IPV4 subnet mask supplied was invalid.

Explanation

The value that you entered for an IPV4 subnet mask is not a valid IPV4 subnet mask.

Action

Specify a valid IPV4 subnet mask, and resubmit the task.

CMMVC6322E The command failed as the IPV4 gateway address supplied was invalid.

Explanation

The value that you entered for an IPV4 gateway address is not a valid IPV4 gateway address.

Action

Specify a valid IPV4 gateway address, and resubmit the task.

CMMVC6323E The command failed as the IPV6 address supplied was invalid.

Explanation

The value that you entered for an IPV6 address is not a valid IPV6 address.

Action

Specify a valid IPV6 address, and resubmit the task.

CMMVC6324E The command failed as the IPV6 prefix supplied was invalid.

Explanation

The value that you entered for an IPV6 address prefix is not a valid IPV6 address prefix.

Action

Specify a valid IPV6 address prefix, and resubmit the task.

CMMVC6325E The command failed as the IPV6 gateway address supplied was invalid.

Explanation

The value that you entered for an IPV6 gateway address is not a valid IPV6 gateway address.

Action

Specify a valid IPV6 gateway address, and resubmit the task.

CMMVC6326E The command failed as the IPV4 service mode address supplied was invalid.

Explanation

The value that you entered for an IPV4 service mode address is not a valid IPV4 service mode address.

Action

Specify a valid IPV4 service mode address, and resubmit the task.

CMMVC6327E The command failed as the IPV6 service mode address supplied was invalid.

Explanation

The value that you entered for an IPV6 service mode address is not a valid IPV6 service mode address.

Action

Specify a valid IPV6 service mode address, and resubmit the task.

CMMVC6328E The command failed as the console address supplied was invalid.

Explanation

The value that you entered for the console address is not a valid console address.

Action

Specify a valid console address, and resubmit the task.

CMMVC6329E The command failed as the SNMP address supplied was invalid.

Explanation

The value that you entered for the SNMP address is not a valid SNMP address.

Action

Specify a valid SNMP address, and resubmit the task.

CMMVC6330E The command failed as an IPV6 SNMP address was specified when the IPV6 protocol stack was unavailable.

Explanation

The IPV6 protocol stack must be available when you submit an IPV6 SNMP address with this task.

Action

Configure the IPV6 protocol stack and ensure that it is available, and resubmit the task.

CMMVC6331E The command failed as an IPV4 SNMP address was specified when the IPV4 protocol stack was unavailable.

Explanation

The IPV4 protocol stack must be available when you submit an IPV4 SNMP address with this task.

Action

Configure the IPV4 protocol stack and ensure that it is available, and resubmit the task.

CMMVC6332E The command failed as an IPV6 email address was specified when the IPV6 protocol stack was unavailable.

Explanation

The IPV6 protocol stack must be available when you submit an IPV6 email address with this task.

Action

Configure the IPV6 protocol stack and ensure that it is available, and resubmit the task.

CMMVC6333E The command failed as an IPV4 email address was specified when the IPV4 protocol stack was unavailable.

Explanation

The IPV4 protocol stack must be available when you submit an IPV4 email address with this task.

Action

Configure the IPV4 protocol stack and ensure that it is available, and resubmit the task.

CMMVC6334E The command failed as the email port number supplied is invalid.

Explanation

The value that you entered for an email port number is not a valid email port number.

Action

Specify a valid email port number, and resubmit the task.

CMMVC6335E The command failed as the combination of parameters provided are either mutually incompatible or would leave the cluster without a functioning protocol stack.

Explanation

You have submitted a task with a combination of parameters and parameter values that is not supported or that does not provide the minimum amount of required information.

Action

Ensure that you specify a supported combination of parameters and parameter values, and resubmit the task.

CMMVC6336E The virtual disk (VDisk) copy was not created because the grain size must be 32, 64, 128 or 256.

Explanation

You have supplied an incorrect value for the -grainsize parameter when you attempted to create a space-efficient VDisk copy.

Action

Specify a supported grain size, and resubmit the command.

CMMVC6337E The action failed because the warning size must be a multiple of 512 bytes.

Explanation

You are attempting to create a space-efficient VDisk copy but you have entered an incorrect value for the -warning parameter. The value can either be a percentage of the VDisk capacity or an absolute value that is a multiple of 512 bytes.

Action

Enter a supported warning value, and resubmit the command.

CMMVC6338E The action failed because the warning size can not be larger than the virtual size.

Explanation

You are attempting to create a space-efficient VDisk copy but you have entered an incorrect value for the -warning parameter. The warning value cannot be greater than the VDisk capacity.

Action

Enter a supported warning value, and resubmit the command.

CMMVC6339E The virtual disk (VDisk) copy was not created because the virtual size was not provided.

Explanation

You are attempting to create an image-mode space-efficient VDisk but you did not set the -size parameter.

Action

Resubmit the command using the -size parameter.

CMMVC6340E The action failed because the value supplied for real size is not a multiple of 512 bytes.

Explanation

You are attempting to create or resize a space-efficient VDisk copy but you have entered an incorrect value for the -rsize parameter. All sizes must be integer multiples of 512 bytes.

Action

Resubmit the command using a supported -rsize parameter value.

CMMVC6341E The action failed because the virtual disk (VDisk) copy is not space-efficient.

Explanation

You are attempting to run a command that is valid only for space-efficient VDisks on a VDisk that is not space-efficient.

Action

Specify a space-efficient VDisk, and resubmit the command.

CMMVC6342E The virtual disk (VDisk) copy was not shrunk because its real size cannot be less than its used size.

Explanation

You are attempting to reduce the real size that is allocated to a space-efficient VDisk copy, but the command cannot be initiated because it would make the real size less than the size that is currently used.

Action

Determine the used size of the VDisk copy, and resubmit the command using a -rsize parameter value that is greater than or equal to the used size.

CMMVC6343E The virtual disk (VDisk) copy was not shrunk because its real size can not be negative.

Explanation

You are attempting to reduce the real size that is allocated to a space-efficient VDisk copy, but the command cannot be initiated because it would make the real size less than zero.

Action

Determine the real size of the VDisk copy, and resubmit the command using a supported -rsize parameter value.

CMMVC6344E The repair operation cannot start because the space-efficient virtual disk (VDisk) copy is already being repaired.

Explanation

You are attempting to repair a space-efficient VDisk copy, but the copy is already being repaired.

Action

Specify the correct vdisk and copy parameters, and resubmit the command.

CMMVC6345E The repair operation cannot start because the space-efficient virtual disk (VDisk) copy was created using -import but the cluster could not recognize its format.

Explanation

You are attempting to repair a space-efficient VDisk copy that is reporting corrupt metadata. The cluster cannot repair the VDisk copy because it was not recognized as a valid space-efficient disk when it was imported into this cluster. The most probable cause is that the wrong MDisk was used when the VDisk copy was imported.

Action

Delete the VDisk copy, and resubmit the import operation using the same MDisk that was exported from the original cluster.

CMMVC6346E The repair operation cannot start because the space-efficient virtual disk (VDisk) copy was created using -import with a real size that is too small.

Explanation

You are attempting to repair a space-efficient VDisk copy that is reporting corrupt metadata. The cluster cannot repair the VDisk copy because although it was recognized as a valid space-efficient disk when it was imported into this cluster, the real size allocated to the VDisk copy is too small. The most probable cause is

that the incorrect value was supplied with -rsize parameter when the VDisk copy was imported.

Action

Delete the VDisk copy. Resubmit the import operation either using a larger value for -rsize, or supplying the -rsize parameter without a value to let the system choose a real size.

CMMVC6347E The specific upgrade package cannot be installed on this hardware level.

Explanation

The version of software that you are attempting to install does not support the hardware level of the configuration node.

Action

Check the release notes for the version of software that you want to install. Ensure that the version of software that you install supports the hardware level of all of the nodes in the cluster, and resubmit the task.

CMMVC6348E The command failed as there was not enough information provided to process successfully.

Explanation

You have submitted a task with a combination of parameters and parameter values that does not provide the minimum amount of required information.

Action

Ensure that you specify a supported combination of parameters and parameter values, and resubmit the task.

CMMVC6349E The command failed because the virtual disk cache has been lost. Use the force flag to acknowledge this.

Explanation

You are attempting to move a VDisk that has lost cache data between I/O groups and did not specifiy the -force flag. You must specify the -force flag when you move a VDisk that has lost cache data between I/O groups.

Action

Specify the -force flag, and resubmit the command.

CMMVC6350E The command failed because there is insufficient mirror bitmap space.

Explanation

The command failed because there is insufficient free memory to allocate the bitmap needed for virtual disk mirroring in the I/O Group.

Action

Perform one of the following actions:

- Submit an svctask chiogrp command to increase the bitmap space.
- Remove virtual disk mirrors from the I/O Group.

Resubmit the command that caused this error.

CMMVC6351E The command failed because the virtual disk (Vdisk) is not mirrored.

Explanation

Only mirrored virtual disks (VDisks) are supported for this command.

Action

Perform one of the following actions:

- Submit the appropriate command for a VDisk that is not mirrored.
- Submit an svctask addvdiskcopy command to add a copy to the virtual disk, and resubmit the command that caused this error.

CMMVC6352E The command failed because the number of copies of this virtual disk (VDisk) would exceed the limit.

Explanation

You cannot exceed the limit on the number of copies that are supported for a virtual disk (VDisk).

Action

Submit an svctask rmvdiskcopy or svctask splitvdiskcopy command to decrease the number of virtual disk copies, and resubmit the command that caused this error.

CMMVC6353E The command failed because the copy specified does not exist.

Explanation

You must specify an existing copy for this command.

Action

Submit an svcinfo lsvdiskcopy command to show all of the available copies for this virtual disk. Select a copy that exists, and then resubmit the command that caused this error.

CMMVC6354E The command failed because a copy is not synchronized.

Explanation

The copy that you specify for this command must be a synchronized copy.

Action

Use the svcinfo lsvdisksyncprogress command to view the synchronization status. Wait for the copy to synchronize. If you want the synchronization process to complete more quickly, increase the rate by submitting an svctask chvdisk command. When the copy is synchronized, resubmit the command that caused this error.

CMMVC6355E The command failed because an image mode copy is not synchronized and -force was not specified.

Explanation

When you specify an image mode copy for this command, the copy must be synchronized unless you also specify the -force parameter.

Action

Perform one of the following actions:

- Use the svcinfo lsvdisksyncprogress command to view the synchronization status. Wait for the copy to synchronize. If you want the synchronization process to complete more quickly, increase the rate by submitting an svctask chvdisk command. When the copy is synchronized, resubmit the command that caused this error.
- Resubmit the command and specify the -force parameter.

NOTE: When you specify the -force parameter with the command that caused this error, the image mode copy is no longer guaranteed to have the correct virtual disk data.

CMMVC6356E The command failed because a copy is not synchronized and -force was not specified.

Explanation

When you specify a copy for this command, the copy must be synchronized unless you also specify the -force parameter.

Action

Perform one of the following actions:

- Use the svcinfo lsvdisksyncprogress command to view the synchronization status. Wait for the copy to synchronize. If you want the synchronization process to complete more quickly, increase the rate by submitting an svctask chvdisk command. When the copy is synchronized, resubmit the command that caused this error.
- Resubmit the command and specify the -force parameter.

NOTE: When you specify the -force parameter with the command that caused this error, the entire virtual disk copy is resynchronized.

CMMVC6357E The command failed because the copy specified is not synchronized and -force was not specified.

Explanation

When you specify a copy for this command, the copy must be synchronized unless you also specify the -force parameter.

Action

Perform one of the following actions:

- Use the svcinfo lsvdisksyncprogress command to view the synchronization status. Wait for the copy to synchronize. If you want the synchronization process to complete more quickly, increase the rate by submitting an svctask chvdisk command. When the copy is synchronized, resubmit the command that caused this error.
- Resubmit the command and specify the -force parameter.

NOTE: When you specify the -force parameter with the command that caused this error, the created virtual disk is no longer guaranteed to have identical data to the original virtual disk when the split is performed.

CMMVC6358E The command failed because the copy specified is the only synchronized copy.

Explanation

The command failed because the copy specified is the only synchronized copy.

Action

Use the svcinfo lsvdisksyncprogress command to view the synchronization status. Wait for another copy to synchronize. If you want the synchronization process to complete more quickly, increase the rate by submitting an svctask chvdisk command. When the copy has synchronized, resubmit the command that caused this error.

CMMVC6359E The command failed because there are insufficient online synchronized copies.

Explanation

This error occurs when at least one of the virtual disk copies is offline.

Action

Fix all of the errors that are associated with the virtual disk copies, and resubmit the command.

CMMVC6363E The command failed because the Logical Block Address (LBA) specified is invalid for this virtual disk (VDisk).

Explanation

You must specify a Logical Block Address (LBA) that is a valid address for this virtual disk (VDisk).

Action

Use the svcinfo lsvdisk command to obtain the virtual disk size, and resubmit the command that caused this error using a logical block address that is in range.

CMMVC6364E The command failed because the logical block address (LBA) requested is too large for the disk.

Explanation

You have specified an LBA in conjunction with a VDisk or MDisk, but the LBA is too large and does not exist on the disk.

Action

Check the size of the disk, and resubmit the command using an LBA that exists on the disk.

CMMVC6365E The command timed out.

Explanation

The command has not completed in a reasonable amount of time. Processing of the command required the software to wait for a set of MDisk reads or writes to complete, and the predefined reasonable wait time has been exceeded.

Action

Resolve any MDisk or fabric error log entries, and resubmit the command.

CMMVC6366E One or more nodes in the cluster has hardware that is not supported by the new software package.

Explanation

The version of software that you are attempting to install does not support the hardware in at least one node in the cluster.

Action

Check the release notes for the version of software that you want to install. Upgrade hardware so that all of the hardware in the cluster is supported by the new version of software, and resubmit the task.

CMMVC6367E A remote cluster is running software that is incompatible with the new software package.

Explanation

The version of software that you are attempting to install on the local cluster does not support the version of software that is installed on the remote cluster.

Action

Check the release notes for the version of software that you want to install. Perform one of the following actions:

- Upgrade the software on the remote cluster to a version that is supported by the version of software that you want to install on the local cluster before you upgrade the software on the local cluster.
- Delete the cluster partnership to stop all remote copy relationships between the clusters, and resubmit the task.

CMMVC6368E The new software package might be incompatible with the remote cluster.

Explanation

The software version compatibility between clusters cannot be checked because the remote cluster is not accessible.

Action

Perform one of the following actions:

- Ensure that the link to the remote cluster is functioning properly, and resubmit the task.
- Delete the cluster partnership to stop all remote copy relationships between the clusters, and resubmit the task.

CMMVC6369E The FlashCopy storage capacity that the cluster is using is approaching the FlashCopy storage capacity that is licensed.

Explanation

You are being warned that the FlashCopy storage capacity license might be exceeded soon.

Action

Upgrade the FlashCopy storage capacity license to prevent recurrence of this warning message.

CMMVC6370E The Remote Copy storage capacity that the cluster is using is approaching the Remote Copy storage capacity that is licensed.

Explanation

You are being warned that the Remote Copy storage capacity license might be exceeded soon.

Action

Upgrade the Remote Copy storage capacity license to prevent recurrence of this warning message.

CMMVC6372E The virtualized storage capacity that the cluster is using is approaching the virtualized storage capacity that is licensed.

Explanation

You are being warned that the virtualized storage capacity license might be exceeded soon.

Action

Upgrade the virtualized storage capacity license to prevent recurrence of this warning message.

CMMVC6373E The virtualized storage capacity that the cluster is using exceeds the virtualized storage capacity that is licensed.

Explanation

You are being warned that the virtualized storage capacity license has been exceeded.

Action

Upgrade the virtualized storage capacity license to prevent recurrence of this warning message.

CMMVC6374E The FlashCopy storage capacity that the cluster is using exceeds the FlashCopy storage capacity that is licensed.

Explanation

You are being warned that the FlashCopy storage capacity license has been exceeded.

Action

Upgrade the FlashCopy storage capacity license to prevent recurrence of this warning message.

CMMVC6375E The Remote Copy storage capacity that the cluster is using exceeds the Remote Copy storage capacity that is licensed.

Explanation

You are being warned that the Remote Copy storage capacity license has been exceeded.

Action

Upgrade the Remote Copy storage capacity license to prevent recurrence of this warning message.

CMMVC6394E The command failed because an attempt to make the virtual disk cache empty took too long.

Explanation

The failed command must empty the virtual disk cache before attempting the requested action to ensure that data is preserved. The empty VDisk cache subtask has taken too long, and therefore the command that you have submitted was not initiated so that other configuration activity can occur.

The system continues attempting to empty the virtual disk cache.

The storage associated with the virtual disk is probably overloaded.

Action

Wait a few minutes to allow the virtual disk cache to empty. Resubmit the command.

Alternatively, you can use the -force parameter, if the command supports the -force parameter, to bypass the empty virtual disk cache subtask. However, specifying the -force parameter will discard cache data for the virtual disk. Only use the -force flag with this command if you do not intend to use the existing contents of the virtual disk.

In addition to the above actions, investigate the performance of the network storage devices associated with this virtual disk. The performance of host applications using these devices might be degraded.

Remedial action to resolve a performance problem enables host application performance to return to optimal conditions, and prevents this error message from recurring when you resubmit the command that caused this error.

CMMVC6399E The command failed because there is not enough memory available for reservation.

Explanation

At least one node in the cluster cannot reserve the required amount of memory. This might be caused by pinned data in the cache.

Action

Check for errors in the error log. Follow the directed maintenance procedures to resolve the problem.

CMMVC6400E The command failed because a specified managed disk (MDisk) is already in use.

Explanation

You cannot specify an MDisk for this command if it is already in a managed disk group or is being used as an image mode VDisk.

Action

Specify an MDisk that is not being used as an image mode VDisk and is not in a managed disk group, and resubmit the command.

CMMVC6401E The command failed because one or more of the specified managed disks (MDisks) that you have specified are not in the required managed disk group.

Explanation

The command requires that all of the MDisks that you specify must be in the same MDisk group.

Action

Ensure that all of the MDisks that you specify are in the same MDisk group, and resubmit the command.

CMMVC6402E The command failed because the managed disk (MDisk) is not in the required managed disk group.

Explanation

All of the MDisks that you specify must be in the required MDisk group. At least one of the source MDisks that you have specified in the command is not in the required MDisk group.

Action

Ensure that all of the MDisks that you specify are in the MDisk group that you specify, and resubmit the command.

CMMVC6403E The command failed because the target managed disk (MDisk) is not in the required managed disk group.

Explanation

All of the MDisks that you specify must be in the required MDisk group. At least one of the target MDisks that you have specified in the command is not in the required MDisk group.

Action

Ensure that all of the MDisks that you specify are in the MDisk group that you specify, and resubmit the command.

CMMVC6404E The command failed because the source and target managed disk groups must be different.

Explanation

The source and target MDisk groups that you specify for a cross MDisk group migration must be different.

Action

Ensure that the source and target MDisk groups that you specify for a cross MDisk group migration are different, and resubmit the command.

CMMVC6405E The command failed because the target copy was not specified.

Explanation

A target copy must be specified when you use migrations on a VDisk and more than one VDisk copy exists.

Action

Specify the target copy, and resubmit the command.

CMMVC6406E The command failed because the specified managed disk group does not exist.

Explanation

At least one of the MDisk groups that you have specified in the parameter list does not exist.

Action

Ensure that each of the MDisk groups that you specify exists, and resubmit the command.

CMMVC6407E The command failed because the managed disk group is invalid.

Explanation

At least one MDisk group ID is above the maximum value that is available for the system.

Action

Ensure that each MDisk group ID that you specify in the parameter list exists, and resubmit the command.

CMMVC6408E The command failed because too few managed disk groups were specified.

Explanation

You must specify the number of MDisk groups that is consistent with the other parameters and parameter values that you specify with the command.

Action

Refer to the command documentation for valid combinations of parameters and parameter values. Use a valid combination of parameters and values, and resubmit the command.

CMMVC6409E The command failed because too many managed disk groups were specified.

Explanation

You must specify the number of MDisk groups that is consistent with the other parameters and parameter values that you specify with the command.

Action

Refer to the command documentation for valid combinations of parameters and parameter values. Use a valid combination of parameters and values, and resubmit the command.

CMMVC6410E The command failed because too few managed disks (MDisks) were specified.

Explanation

You must specify the number of MDisks that is consistent with the other parameters and parameter values that you specify with the command.

Action

Refer to the command documentation for valid combinations of parameters and parameter values. Use a valid combination of parameters and values, and resubmit the command.

CMMVC6411E The command failed because too many managed disks (MDisks) were specified.

Explanation

You must specify the number of MDisks that is consistent with the other parameters and parameter values that you specify with the command.

Action

Refer to the command documentation for valid combinations of parameters and parameter values. Use a valid combination of parameters and values, and resubmit the command.

CMMVC6412E The command failed because the managed disk group extent size is above maximum permitted size.

Explanation

You cannot specify an MDisk group extent size that is larger the maximum size.

Action

Specify an MDisk group extent size that is less than or equal to the maximum size, and resubmit the command.

CMMVC6413E The command failed because the managed disk (MDisk) is invalid.

Explanation

At least one MDisk ID is above the maximum value that is available for the system.

Action

Ensure that each MDisk ID that you specify in the parameter list exists, and resubmit the command.

CMMVC6414E The command failed because the managed disk (MDisk) is currently being migrated.

Explanation

When you submit this command, you cannot specify an MDisk that is being migrated.

Action

Either wait until the migration has completed for the MDisk that you specify, or specify a different MDisk, and resubmit the command.

CMMVC6415E The command failed because the managed disk group warning threshold is too low.

Explanation

You must specify an MDisk group warning threshold that is equal to or greater than the minimum size.

Action

Specify an MDisk group warning threshold that is equal to or greater than the minimum size, and resubmit the command.

CMMVC6416E The command failed because the managed disk group warning threshold is too high.

Explanation

You must specify an MDisk group warning threshold size that is equal to or less than the size of the MDisk group when all of the MDisks have been added, or you must specify an MDisk group warning percentage that is equal to or less than the maximum warning threshold percentage.

Action

Specify valid values for the MDisk group warning threshold size or percentage, and resubmit the command.

CMMVC6417E The command failed because the managed disk group warning threshold is invalid.

Explanation

To specify the warning threshold there must be at least one managed MDisk in the MDisk Group.

Action

Ensure that there is at least one MDisk defined for the MDisk group or remove the warning threshold, and resubmit the command.

CMMVC6418E The command failed because the virtual disk (VDisk) is in the process of being resized.

Explanation

When you submit this command, you cannot specify a VDisk that is being resized.

Action

Wait for the resize VDisk operation to complete. If you still want to submit this command after the operation has completed, resubmit the command.

CMMVC6419E The command failed because one or more of the specified managed disks (MDisks) are in the process of being deleted.

Explanation

When you submit this command, you cannot specify an MDisk that is being deleted with the -force option.

Action

Wait for the delete MDisk operation to complete. Do not include any MDisks that have been deleted in the list of MDisks that you specify, and resubmit the command.

CMMVC6423E The Send Inventory email operation failed because email is not started.

Explanation

The send inventory email functionality has been enabled but the email service has not been started.

Action

Disable the send inventory email functionality or start the email service.

CMMVC6424E The Send Inventory email operation failed because there are no inventory email users.

Explanation

The send inventory functionality has been enabled but no email users with the ability to receive inventory emails have been created.

Action

Either turn off the send inventory email functionality or create an email user account that is capable of receiving inventory emails. Refer to the documentation for the mkemailuser command for help on creating email users.

CMMVC6425E The action failed because the maximum number of objects has been reached.

Explanation

The action failed because the maximum number of objects has been reached.

Action

Not applicable.

CMMVC6426E The command failed because a specified managed disk (MDisk) is already in use.

Explanation

You cannot specify an MDisk that is already configured as an image mode VDisk.

Action

Specify an unmanaged disk, and resubmit the task.

CMMVC6427E The command failed because one or more of the specified managed disks (MDisks) are not in the required managed disk group.

Explanation

The create VDisk task requires that all of the MDisks that you specify must be in the same MDisk group.

Action

Ensure that all of the MDisks that you specify are in the same MDisk group, and resubmit the task.

CMMVC6428E The command failed because the source managed disk (MDisk) is not in the required managed disk group.

Explanation

The task requires that all of the source MDisks that you specify must be in the same MDisk group.

Action

Ensure that all of the source MDisks that you specify are in the same MDisk group, and resubmit the task.

CMMVC6429E The command failed because the target managed disk (MDisk) is not in the required managed disk group.

Explanation

The task requires that all of the target MDisks that you specify must be in the same MDisk group.

Action

Ensure that all of the target MDisks that you specify are in the same MDisk group, and resubmit the task.

CMMVC6430E The command failed because the target and source managed disk groups must be different.

Explanation

The cross MDisk group migration task does not support specifying the same MDisk group to be both the source and target MDisk group.

Action

Specify a source MDisk group and a target MDisk group that are not identical, and resubmit the task.

CMMVC6431E The command failed because the target copy was not specified.

Explanation

When you use migrations on a VDisk and there is more than one copy, you must specify which copy to use as the target copy.

Action

Specify the target copy, and resubmit the task.

CMMVC6432E The command failed because the specified managed disk group does not exist.

Explanation

All of the MDisk groups that you specify must already exist.

Action

Ensure that all of the MDisk groups that you specify already exist, and resubmit the task.

CMMVC6433E The command failed because the managed disk group is invalid.

Explanation

All of the MDisk group IDs that you specify must have a value that is less than or equal to the maximum supported MDisk group ID value.

Action

Ensure that all MDisk groups have supported ID values. Ensure that all of the MDisk groups that you specify already exist, and resubmit the task.

CMMVC6434E The command failed because too few managed disk groups were specified.

Explanation

The combination of parameters and parameter values that you have specified is not supported. The task requires that you specify more MDisk groups than the number that you have specified.

Action

Specify a supported combination of parameters and parameter values, and resubmit the task.

CMMVC6435E The command failed because too many managed disk groups were specified.

Explanation

The combination of parameters and parameter values that you have specified is not supported. The task requires that you specify fewer MDisk groups than the number that you have specified.

Action

Specify a supported combination of parameters and parameter values, and resubmit the task.

CMMVC6436E The command failed because too few managed disks (MDisks) were specified.

Explanation

The combination of parameters and parameter values that you have specified is not supported. The task requires that you specify more MDisks than the number that you have specified.

Action

Specify a supported combination of parameters and parameter values, and resubmit the task.

CMMVC6437E The command failed because too many managed disks (MDisks) were specified.

Explanation

The combination of parameters and parameter values that you have specified is not supported. The task requires that you specify fewer MDisks than the number that you have specified.

Action

Specify a supported combination of parameters and parameter values, and resubmit the task.

CMMVC6438E The command failed because the managed disk group extent size is above maximum permitted size.

Explanation

The MDisk group extent size that you have specified is greater than the supported maximum value.

Action

Specify a supported MDisk group extent size, and resubmit the task.

CMMVC6439E The command failed because the managed disk (MDisk) is invalid.

Explanation

Each MDisk ID must have a value that is less than or equal to the maximum supported MDisk ID value.

Action

Ensure that all of the MDisks have supported ID values. Ensure that all of the MDisks that you specify already exist, and resubmit the task.

CMMVC6440E The command failed because the managed disk (MDisk) is currently being migrated.

Explanation

When you submit this task, you cannot specify an MDisk that is being migrated.

Action

Ensure that the MDisk that you specify is not migrating, and resubmit the task. If you want to specify the same MDisk and resubmit the task, ensure that the migration for that MDisk has completed before you resubmit the task.

CMMVC6441E The command failed because the managed disk group warning threshold is too low.

Explanation

The value that you have specified for the MDisk group warning threshold is less than the minimum supported value.

Action

Specify a supported value for the MDisk group warning threshold, and resubmit the task.

CMMVC6442E The command failed because the managed disk group warning threshold is too high.

Explanation

Either the value for the MDisk group warning percentage is greater than the maximum supported value, or the MDisk group warning disk size is greater than the MDisk group capacity.

Action

Specify supported values for MDisk group warning percentage and disk size, and resubmit the task.

CMMVC6443E The command failed because the managed disk group warning threshold is invalid.

Explanation

If you submit this command and specify an MDisk group warning threshold percentage, you must specify an MDisk group that contains at least one MDisk and you must specify a supported value for the MDisk group warning threshold percentage.

Action

Either do not specify an MDisk group warning threshold percentage, or specify a supported value for the MDisk group warning threshold percentage and specify an MDisk group that contains at least one MDisk, and resubmit the task.

CMMVC6444E The command failed because the virtual disk (VDisk) is in the process of being resized.

Explanation

You cannot specify a VDisk that is being resized when you submit this task.

Action

Wait for the resize VDisk task to complete. You can specify the same VDisk and resubmit this task only after the resize VDisk task that is in progress has completed.

CMMVC6445E The command failed because one or more of the specified managed disks (MDisks) are in the process of being deleted.

Explanation

You cannot specify an MDisk that is being force deleted.

Action

Wait until all force delete MDisk tasks have completed. Ensure that all of the MDisks that you specify still exist, and resubmit the task.

CMMVC6446E The command failed because the managed disk groups have different extent sizes.

Explanation

This task requires that the extent size of the source MDisk group and the extent size of the target MDisk group must be identical.

Action

If you want to resubmit this command, ensure that the source and target MDisk groups have the same extent size. If you want to move a VDisk to an MDisk group

that has a different extent size, you must use the procedure that is documented in the technical notes.

CMMVC6447E The command failed because the virtual disk (VDisk) is currently being migrated.

Explanation

You cannot specify a VDisk that is being migrated.

Action

Either wait until the VDisk migration process has completed and resubmit the task, or specify a VDisk that is not being migrated and resubmit the task.

CMMVC6448E Deleting this node will cause data loss for resources associated with the IO group of this node.

Explanation

This node contains resources which are vital to the IO group and unavailable elsewhere. Removing this node will cause a loss of customer data.

It is recommended that this node not be removed unless the customer data supported by it is of no consequence.

Action

The -force option must be used to remove this node.

CMMVC6449E The operation was not performed because the partnership owns Global or Metro Mirror relationships or consistency groups.

Explanation

The cluster partnership cannot be removed while there are Global or Metro Mirror relationships or consistency groups that are configured in the local cluster and that are associated with the remote cluster of the partnership.

Action

Identify all of the Global or Metro Mirror relationships or consistency groups in the local cluster that are configured between this cluster and the remote cluster of the partnership. Remove all of the relationships and groups that you have identified, and resubmit the task. NOTE: Do not remove relationships or groups that are associated with a different cluster, and do not remove relationships or groups that are contained entirely within the local cluster.

CMMVC6450W A FlashCopy relationship was created but physical_flash is not enabled.

Explanation

The create FlashCopy relationship task has succeeded. However, physical_flash should be enabled when you create a FlashCopy relationship in the physical disk license scheme.

Action

Ensure that you have the appropriate virtualization license for the cluster configuration that you want to enable. Ensure that the license settings for this cluster match the license.

Delete the FlashCopy relationship or enable physical_flash.

CMMVC6451W A Global Mirror or Metro Mirror relationship was created but physical_remote is not enabled.

Explanation

The create Global Mirror or Metro Mirror relationship task has succeeded. However, physical_remote should be enabled when you create a Global Mirror or Metro Mirror relationship and the cluster uses the physical disk license scheme.

Action

Ensure that you have the appropriate virtualization license for the cluster configuration that you want to enable. Ensure that the license settings for this cluster match the license.

Delete the Global Mirror or Metro Mirror relationship or enable physical_remote.

CMMVC6452W You are using the physical disk license scheme but the values for physical_flash and physical_remote are not set.

Explanation

The task has succeeded. However, you should enable physical_flash before you create a FlashCopy relationship and you should enable physical_remote before you create a Global Mirror or Metro Mirror relationship.

Action

Enable physical_flash before you create a FlashCopy relationship. Enable physical_remote before you create a Global Mirror or Metro Mirror relationship.

CMMVC6453W You have disabled the physical disk license scheme but the capacity license scheme is not set.

Explanation

The task has succeeded. However, you should configure a license scheme before you create a FlashCopy, Global Mirror or Metro Mirror relationship. You can configure a physical disk license scheme or a capacity license scheme, but not both.

Action

If you do not have a virtualization feature license that is valid for this cluster, contact your IBM sales representative and obtain a license. Ensure that the license settings for this cluster match the license that you have for this cluster.

CMMVC6454E The command failed because the physical disk license scheme is not enabled.

Explanation

You can only enable physical_flash or physical_remote when the physical disk license scheme is enabled.

Action

Ensure that you have the appropriate virtualization license for the cluster configuration that you want to enable. Ensure that the license settings for this cluster match the license. Resubmit the task if it supported by the license.

CMMVC6455E The command failed because a capacity license scheme parameter was specified but the physical disk license scheme is enabled.

Explanation

You cannot enable the capacity license scheme or specify a capacity license scheme parameter while the cluster is using the physical disk license scheme.

Action

Ensure that you have the appropriate virtualization license for the cluster configuration that you want to enable. Ensure that the license settings for this cluster match the license. Resubmit the task if it supported by the license.

CMMVC6456E The command failed because a physical disk license scheme parameter was specified but the capacity license scheme is enabled.

Explanation

You cannot enable the physical disk license scheme or specify a physical disk license scheme parameter while the cluster is using the capacity license scheme.

Action

Ensure that you have the appropriate virtualization license for the cluster configuration that you want to enable. Ensure that the license settings for this cluster match the license. Resubmit the task if it supported by the license.

CMMVC6457E One or more quorum disks are on the specified controller.

Explanation

You cannot disable the setting that allows a controller to support a quorum disk while a quorum disk is configured on the controller.

Action

Move all quorum disks from the controller to a different storage system using the svctask setquorum command, and resubmit this task.

CMMVC6458E The specified controller cannot support quorum disks.

Explanation

The controller type of the controller that you specified does not support quorum disks.

Action

Specify a controller that has a controller type that supports quorum disks, and resubmit the task.

CMMVC6459E The mkrcrelationship command failed because the same VDisk was specified as the master and auxiliary VDisk.

Explanation

A relationship cannot be created from a VDisk to itself. The mkrcrelationship command requires that you specify two different VDisks for the master and auxiliary positions. These can be two VDisks in the local cluster, or a VDisk in each of two different clusters.

Action

Specify a master VDisk and an auxiliary VDisk that are not identical to each other, and resubmit the task.

CMMVC6460E The command failed because the migration source is offline.

Explanation

The source of the migration is offline. The offline source is either an image mode MDisk or the entire managed disk group.

Action

- If you submitted the rmmdisk command and specified a regular MDisk, determine the managed disk group to which the source MDisk is defined, and follow the procedure for bringing the managed disk group online. There will be an entry in the error log for the corresponding managed disk group.
- If you submitted the rmmdisk command and specified an image mode MDisk, determine the source MDisk and follow the procedure for bringing the image mode MDisk online. There will be an entry in the error log for the corresponding MDisks.
- If you submitted a command to migrate a copy of an image mode VDisk, determine the corresponding source MDisk and follow the procedure for diagnosing problems with MDisks. There will be an entry in the error log for the corresponding MDisks.
- If you submitted any other command to migrate a VDisk copy, determine the managed disk group to which the VDisk is defined, and follow the procedure for bringing the managed disk group online. There will be an entry in the error log for the corresponding managed disk group.

CMMVC6461E The command failed because starting the migration will result in VDisks going offline in the source managed disk group.

Explanation

A migration from an image mode VDisk will use the source managed disk group and the source managed disk group assumes the combined state of the image mode MDisk and the managed disk group. If the online or offline states of the image mode MDisk and the managed disk group are different on different nodes, the source VDisk might go offline or all of the VDisks in the source managed disk group might go offline.

Action

For each node, note the online or offline states of the source VDisk and the source managed disk group. If one entity is online and the other is offline, bring online whichever is offline. Taking the online entity offline is not recommended because other VDisks might go offline.

CMMVC6462E The command failed because starting the migration will result in VDisks going offline because the target managed disk group is offline.

Explanation

The migration process assigns the VDisk an online or offline state based on the states of the source and target managed disk groups. In this case, based on the offline state of the target managed disk group the VDisk that is currently online would have been taken offline. The command cannot be initiated because this action is not supported. There will be an entry in the error log for the corresponding managed disk group.

Action

For each node, note the online or offline state of the source and target managed disk groups. For each node, if one of these two managed disk groups is online and

the other is offline, bring online whichever managed disk group is offline. Taking the online managed disk group offline is not recommended because other VDisks might go offline.

CMMVC6463E The command failed because Starting the migration will result in VDisks going offline because a target MDisk is offline.

Explanation

The VDisk is currently online. The migration process assigns the VDisk an online or offline state based on the states of the source and target MDisks. In this case, based on the offline state of the target MDisk, the VDisk would have been taken offline. The task cannot be initiated because this action is not supported.

Action

Bring the target MDisk online by following the recommended procedure for bringing an MDisk online, and resubmit the command.

CMMVC6500E The action failed because the source and destination virtual disk (VDisk) are the same.

Explanation

The action failed because the source and destination vdisk are the same.

CMMVC6501E The action failed because the node hardware is incompatible with the current I/O group member.

Explanation

The action failed because the node hardware is incompatible with the current I/O group member.

CMMVC6502E The FlashCopy mapping was not prepared because preparing consistency group 0 is not a valid operation.

Explanation

The FlashCopy mapping was not prepared because preparing consistency group 0 is not a valid operation.

CMMVC6503E The FlashCopy mapping or consistency group was not stopped because stopping consistency group 0 is not a valid operation.

Explanation

The FlashCopy mapping or consistency group was not stopped because stopping consistency group 0 is not a valid operation.

Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully.

Features

These are the major accessibility features in the SAN Volume Controller Console:

- You can use screen-reader software and a digital speech synthesizer to hear what is displayed on the screen. The following screen reader has been tested: Window-Eyes v6.1.
- You can operate all features using the keyboard instead of the mouse.
- You can change the initial delay and repeat rate of the up and down buttons to two seconds when you use the front panel of the SAN Volume Controller to set or change an IPv4 address. This feature is documented in the applicable sections of the SAN Volume Controller publications.

Navigating by keyboard

You can use keys or key combinations to perform operations and initiate many menu actions that can also be done through mouse actions. You can navigate the SAN Volume Controller Console and help system from the keyboard by using the following key combinations:

- To traverse to the next link, button, or topic, press Tab inside a frame (page).
- To expand or collapse a tree node, press → or ←, respectively.
- To move to the next topic node, press V or Tab.
- To move to the previous topic node, press ^ or Shift+Tab.
- To scroll all the way up or down, press Home or End, respectively.
- To go back, press Alt+←.
- To go forward, press Alt+→.
- To go to the next frame, press Ctrl+Tab.
- To move to the previous frame, press Shift+Ctrl+Tab.
- To print the current page or active frame, press Ctrl+P.
- To select, press Enter.

Accessing the publications

You can view the publications for the SAN Volume Controller in Adobe Portable Document Format (PDF) using the Adobe Acrobat Reader. The PDFs are provided at the following Web site:

www.ibm.com/storage/support/2145

Related reference

"SAN Volume Controller library and related publications" on page xxvii
A list of other publications that are related to this product are provided to you for your reference.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATIONS "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Almaden Research
650 Harry Road
Bldg 80, D3-304, Department 277
San Jose, CA 95120-6099
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products may be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application

programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe and the Adobe logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Intel logo, Intel Xeon, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Glossary

This glossary includes terms for the IBM System Storage SAN Volume Controller.

This glossary includes selected terms and definitions from A Dictionary of Storage Networking Terminology (www.snia.org/education/dictionary), copyrighted 2001 by the Storage Networking Industry Association, 2570 West El Camino Real, Suite 304, Mountain View, California 94040-1313. Definitions derived from this book have the symbol (S) after the definition.

The following cross-references are used in this glossary:

See Refers the reader to one of two kinds of related information:

- A term that is the expanded form of an abbreviation or acronym. This expanded form of the term contains the full definition.
- A synonym or more preferred term.

See also
Refers the reader to one or more related terms.

Contrast with
Refers the reader to a term that has an opposite or substantively different meaning.

Numerics

2145 A hardware machine type for the IBM System Storage SAN Volume Controller. Models of the SAN Volume Controller are expressed as the number 2145 followed by "-xxx", such as 2145-8G4. Hardware models for the 2145 include 2145-4F2, 2145-8F2, 2145-8F4, 2145-8G4, and 2145-8A4.

A

access mode
One of three different modes in which a logical unit (LU) in a disk controller system can operate. See also *image mode*, *managed space mode*, and *unconfigured mode*.

Address Resolution Protocol (ARP)
A protocol that dynamically maps an IP address to a network adapter address in a local area network.

agent code
An open-systems standard that interprets Common Information Model (CIM) requests and responses as they transfer between the client application and the device.

application server
A host that is attached to the storage area network (SAN) and that runs applications.

ARP See *Address Resolution Protocol*.

array An ordered collection, or group, of physical storage devices that are used to define logical volumes or devices.

association
A class that contains two references that define a relationship between two referenced objects.

asymmetric virtualization
A virtualization technique in which the virtualization engine is outside the data path and performs a metadata-style service. The metadata server contains all the mapping and locking tables while the storage devices contain only data. See also *symmetric virtualization*.

auxiliary virtual disk
The virtual disk that contains a backup copy of the data and that is used in disaster recovery scenarios. See also *master virtual disk*.

availability
The ability of a system to continue working, with perhaps a decrease in performance, after individual components fail.

B

bandwidth
The range of frequencies an electronic system can transmit or receive. The greater the bandwidth of a system, the more information the system can transfer in a given period of time.

bitmap
A coded representation in which each bit, or group of bits, represents or corresponds to an item; for example, a configuration of bits in main storage in which each bit indicates whether a peripheral device or a storage block is available or in which each group of bits corresponds to one pixel of a display image.

blade One component in a system that is designed to accept some number of components (blades). Blades could be individual servers that plug into a multiprocessing system or individual port cards that add connectivity to a switch. A blade is typically a hot-swappable hardware device.

block A unit of data storage on a disk drive.

block virtualization
The act of applying virtualization to one or more block-based (storage) services for the purpose of providing a new aggregated, higher-level, richer, simpler, or secure block service to clients. Block virtualization functions can be nested. A disk drive, RAID system, or volume manager all perform some form of block-address to (different) block-address mapping or aggregation. See also *virtualization*.

Boolean
Pertaining to the processes used in the algebra formulated by George Boole.

C

cache A high-speed memory or storage device used to reduce the effective time required to read data from or write data to lower-speed memory or a device. Read cache holds data in anticipation that it will be requested by a client. Write cache holds data written by a client until it can be safely stored on more permanent storage media such as disk or tape.

Call Home
In SAN Volume Controller, a communication service that sends data and

event notifications to a service provider. The machine can use this link to place a call to IBM or to another service provider when service is required.

capacity licensing
A type of licensing that grants you the use of a number of terabytes (TB) for virtualization, a number of terabytes for Metro Mirror and Global Mirror relationships, and a number of terabytes for FlashCopy mappings.

cascading
The process of connecting two or more fibre-channel hubs or switches together to increase the number of ports or extend distances.

CIM See *Common Information Model*.

CIM object manager (CIMOM)
The common conceptual framework for data management that receives, validates, and authenticates the CIM requests from the client application. It then directs the requests to the appropriate component or service provider.

CIMOM
See *CIM object manager*.

class The definition of an object within a specific hierarchy. A class can have properties and methods and can serve as the target of an association.

CLI See *command line interface*.

client A computer system or process that requests a service of another computer system or process that is typically referred to as a server. Multiple clients can share access to a common server.

client application
A storage management program that initiates Common Information Model (CIM) requests to the CIM agent for the device.

cluster
In SAN Volume Controller, up to four pairs of nodes that provide a single configuration and service interface.

command line-interface (CLI)
A type of computer interface in which the input command is a string of text characters.

Common Information Model (CIM)
A set of standards developed by the Distributed Management Task Force (DMTF). CIM provides a conceptual framework for storage management and an open approach to the design and implementation of storage systems, applications, databases, networks, and devices.

concurrent maintenance
Service that is performed on a unit while it is operational.

In SAN Volume Controller, the ability for one node in the cluster to be turned off for maintenance without interrupting access to the VDisk data provided by the cluster.

configuration node
A node that acts as the focal point for configuration commands and manages the data that describes the cluster configuration.

connected
In a Global Mirror relationship, pertaining to the status condition that occurs when two clusters can communicate.

consistency group
A group of copy relationships between virtual disks that are managed as a single entity.

consistent copy
In a Metro or Global Mirror relationship, a copy of a secondary virtual disk (VDisk) that is identical to the primary VDisk from the viewpoint of a host system, even if a power failure occurred while I/O activity was in progress.

consistent-stopped
In a Global Mirror relationship, the state that occurs when the secondary virtual disk (VDisk) contains a consistent image, but the image might be out-of-date with respect to the primary VDisk. This state can happen when a relationship was in the consistent-synchronized state when an error occurred that forced a freeze of the consistency group. This state can also happen when a relationship is created with the create-consistent flag set to `TRUE`.

consistent-synchronized
In a Global Mirror relationship, the status condition that occurs when the primary virtual disk (VDisk) is accessible for read and write I/O operations. The secondary VDisk is accessible for read-only I/O operations. See also *primary virtual disk* and *secondary virtual disk*.

container
A data storage location; for example, a file, directory, or device.

A software object that holds or organizes other software objects or entities.

contingency capacity
Initially, a fixed amount of unused real capacity that is maintained on a space-efficient virtual disk that is configured to automatically expand its real capacity. It is also the difference between the used capacity and the new real capacity when the real capacity is changed manually.

copied
In a FlashCopy mapping, a state that indicates that a copy has been started after the copy relationship was created. The copy process is complete and the target disk has no further dependence on the source disk.

copying
A status condition that describes the state of a pair of virtual disks (VDisks) that have a copy relationship. The copy process has been started but the two virtual disks are not yet synchronized.

Copy Services
The services that enable you to copy virtual disks (VDisks): FlashCopy, Metro, and Global Mirror.

counterpart SAN
A nonredundant portion of a redundant storage area network (SAN). A counterpart SAN provides all the connectivity of the redundant SAN but without the redundancy. Each counterpart SANs provides an alternate path for each SAN-attached device. See also *redundant SAN*.

cross-volume consistency
In SAN Volume Controller, a consistency group property that guarantees consistency between virtual disks when an application issues dependent write operations that span multiple virtual disks.

D

data migration
The movement of data from one physical location to another without disrupting I/O operations.

degraded
Pertaining to a valid configuration that has suffered a failure but continues to be supported and legal. Typically, a repair action can be performed on a degraded configuration to restore it to a valid configuration.

dense wavelength division multiplexing (DWDM)
A technology that places many optical signals onto one single-mode fiber using slightly different optical frequencies. DWDM enables many data streams to be transferred in parallel.

dependent write operations
A set of write operations that must be applied in the correct order to maintain cross-volume consistency.

destage
A write command initiated by the cache to flush data to disk storage.

device In the CIM Agent, the storage server that processes and hosts client application requests.

IBM definition: A piece of equipment that is used with the computer and does not generally interact directly with the system, but is controlled by a controller.

HP definition: In its physical form, a magnetic disk that can be attached to a SCSI bus. The term is also used to indicate a physical device that has been made part of a controller configuration; that is, a physical device that is known to the controller. Units (virtual disks) can be created from devices after the devices have been made known to the controller.

device provider
A device-specific handler that serves as a plug-in for the Common Information Model (CIM); that is, the CIM object manager (CIMOM) uses the handler to interface with the device.

directed maintenance procedures
The set of maintenance procedures that can be run for a cluster. These procedures are run from within the SAN Volume Controller application and are documented in the *IBM System Storage SAN Volume Controller Troubleshooting Guide*.

disconnected
In a Metro or Global Mirror relationship, pertains to two clusters when they cannot communicate.

discovery
The automatic detection of a network topology change, for example, new and deleted nodes or links.

disk controller
A device that coordinates and controls the operation of one or more disk drives and synchronizes the operation of the drives with the operation of the system as a whole. Disk controllers provide the storage that the cluster detects as managed disks (MDisks).

disk drive
A disk-based, nonvolatile, storage medium.

disk zone
A zone defined in the storage area network (SAN) fabric in which the SAN Volume Controller can detect and address the logical units that the disk controllers present.

Distributed Management Task Force (DMTF)
An organization that defines standards for the management of distributed systems. See also *Common Information Model*.

DMP See *directed maintenance procedures*.

DMTF
See *Distributed Management Task Force*.

domain name server
In the Internet suite of protocols, a server program that supplies name-to-address conversion by mapping domain names to IP addresses.

DRAM
See *dynamic random access memory*.

DWDM
See *dense wavelength division multiplexing*.

dynamic random access memory (DRAM)
A storage in which the cells require repetitive application of control signals to retain stored data.

E

EC See *engineering change.*

EIA See *Electronic Industries Alliance*.

Electronic Industries Alliance (EIA)
An alliance of four trade associations: The Electronic Components, Assemblies & Materials Association (ECA); the Government Electronics and Information Technology Association (GEIA); the JEDEC Solid State Technology Association (JEDEC); and the Telecommunications Industry Association (TIA). Prior to 1998, EIA was the Electronic Industries Association and the group dates back to 1924.

empty In a Global Mirror relationship, a status condition that exists when the consistency group contains no relationships.

engineering change (EC)
A correction for a defect of hardware or software that is applied to a product.

error code
A value that identifies an error condition.

ESS See *IBM TotalStorage Enterprise Storage Server®*.

exclude
To remove a managed disk (MDisk) from a cluster because of certain error conditions.

excluded
In SAN Volume Controller, the status of a managed disk that the cluster has removed from use after repeated access errors.

extent A unit of data that manages the mapping of data between managed disks and virtual disks.

F

fabric In fibre-channel technology, a routing structure, such as a switch, that receives addressed information and routes it to the appropriate destination. A fabric can consist of more than one switch. When multiple fibre-channel switches are interconnected, they are described as cascading. See also *cascading*.

fabric port (F_port)
A port that is part of a fibre-channel fabric. An F_port on a fibre-channel fabric connects to the node port (N_port) on a node.

failover
In SAN Volume Controller, the function that occurs when one redundant part of the system takes over the workload of another part of the system that has failed.

FCIP See *Fibre Channel over IP*.

fibre channel
A technology for transmitting data between computer devices at a data rate of up to 4 Gbps. It is especially suited for attaching computer servers to shared storage devices and for interconnecting storage controllers and drives.

fibre-channel extender
A device that extends a fibre-channel link over a greater distance than is supported by the standard, usually a number of miles or kilometers. Devices must be deployed in pairs at each end of a link.

Fibre Channel over IP (FCIP)
A network storage technology that combines the features of the Fibre Channel Protocol and the Internet Protocol (IP) to connect distributed SANs over large distances.

Fibre Channel Protocol (FCP)
A protocol that is used in fibre-channel communications with five layers that define how fibre-channel ports interact through their physical links to communicate with other ports.

field replaceable unit (FRU)
An assembly that is replaced in its entirety when any one of its components fails. An IBM service representative performs the replacement. In some cases, a field replaceable unit might contain other field replaceable units.

FlashCopy mapping
A relationship between two virtual disks.

FlashCopy relationship
See *FlashCopy mapping*.

FlashCopy service
In SAN Volume Controller, a copy service that duplicates the contents of a source virtual disk (VDisk) to a target VDisk. In the process, the original contents of the target VDisk are lost. See also *point-in-time copy.*

F_port See *fabric port*.

FRU See *field replaceable unit*.

G

gateway
An entity that operates above the link layer and translates, when required, the interface and protocol used by one network into those used by another distinct network.

GB See *gigabyte*.

GBIC See *gigabit interface converter*.

gigabit interface converter (GBIC)
An interface module that converts the light stream from a fibre-channel cable into electronic signals for use by the network interface card.

gigabyte (GB)
In decimal notation, 1 073 741 824 bytes.

Global Mirror
An asynchronous copy service that enables host data on a particular source virtual disk (VDisk) to be copied to the target VDisk that is designated in the relationship.

grain In a FlashCopy bitmap, the unit of data represented by a single bit.

graphical user interface (GUI)
A type of computer interface that presents a visual metaphor of a real-world scene, often of a desktop, by combining high-resolution graphics, pointing devices, menu bars and other menus, overlapping windows, icons and the object-action relationship.

GUI See *graphical user interface*.

H

hardcoded
Pertaining to software instructions that are statically encoded and not intended to be altered.

HBA See *host bus adapter*.

HLUN
See *virtual disk*.

hop One segment of a transmission path between adjacent nodes in a routed network.

host An open-systems computer that is connected to the SAN Volume Controller through a fibre-channel interface.

host bus adapter (HBA)
In SAN Volume Controller, an interface card that connects a host bus, such as a peripheral component interconnect (PCI) bus, to the storage area network.

host ID
In SAN Volume Controller, a numeric identifier assigned to a group of host fibre-channel ports for the purpose of logical unit number (LUN) mapping. For each host ID, there is a separate mapping of Small Computer System Interface (SCSI) IDs to virtual disks (VDisks).

host zone
A zone defined in the storage area network (SAN) fabric in which the hosts can address the SAN Volume Controllers.

hub A fibre-channel device that connects nodes into a logical loop by using a physical star topology. Hubs will automatically recognize an active node and insert the node into the loop. A node that fails or is powered off is automatically removed from the loop.

A communications infrastructure device to which nodes on a multi-point bus or loop are physically connected. Commonly used in Ethernet and fibre-channel networks to improve the manageability of physical cables. Hubs maintain the logical loop topology of the network of which they are a part, while creating a "hub and spoke" physical star layout. Unlike switches, hubs do not aggregate bandwidth. Hubs typically support the addition or removal of nodes from the bus while it is operating. (S) Contrast with *switch*.

I

IBM System Storage Productivity Center (SSPC)
An integrated hardware and software solution that provides a single point of entry for managing SAN Volume Controller clusters, IBM System Storage DS8000™ systems, and other components of a data storage infrastructure.

IBM TotalStorage Enterprise Storage Server (ESS)
An IBM product that provides an intelligent disk-storage system across an enterprise.

ID See *identifier*.

identifier (ID)
A sequence of bits or characters that identifies a user, program device, or system to another user, program device, or system.

idle In a FlashCopy mapping, the state that occurs when the source and target virtual disks (VDisks) act as independent VDisks even if a mapping exists between the two. Read and write caching is enabled for both the source and the target.

idling The status of a pair of virtual disks (VDisks) that have a defined copy relationship for which no copy activity has yet been started.

In a Metro or Global Mirror relationship, the state that indicates that the master virtual disks (VDisks) and auxiliary VDisks are operating in the primary role. Consequently, both VDisks are accessible for write I/O operations.

idling-disconnected
In a Global Mirror relationship, the state that occurs when the virtual disks (VDisks) in this half of the consistency group are all operating in the primary role and can accept read or write I/O operations.

illegal configuration
A configuration that will not operate and will generate an error code to indicate the cause of the problem.

image mode
An access mode that establishes a one-to-one mapping of extents in the managed disk (MDisk) with the extents in the virtual disk (VDisk). See also *managed space mode* and *unconfigured mode*.

image VDisk
A virtual disk (VDisk) in which there is a direct block-for-block translation from the managed disk (MDisk) to the VDisk.

IML See *initial microcode load*.

inconsistent
In a Metro or Global Mirror relationship, pertaining to a secondary virtual disk (VDisk) that is being synchronized with the primary VDisk.

inconsistent-copying
In a Global Mirror relationship, the state that occurs when the primary virtual disk (VDisk) is accessible for read and write input/output (I/O) operations, but the secondary VDisk is not accessible for either. This state occurs after a **start** command is issued to a consistency group that is in the inconsistent-stopped state. This state also occurs when a **start** command is issued, with the force option, to a consistency group that is in the idling or consistent-stopped state.

inconsistent-disconnected
In a Global Mirror relationship, a state that occurs when the virtual disks (VDisks) in the half of the consistency group that is operating in the secondary role are not accessible for either read or write I/O operations.

inconsistent-stopped
In a Global Mirror relationship, the state that occurs when the primary virtual disk (VDisk) is accessible for read and write input/output (I/O) operations, but the secondary VDisk is not accessible for either read or write I/O operations.

indication
An object representation of an event.

initial microcode load (IML)
In SAN Volume Controller, the process by which the run-time code and data for a node are loaded into memory and initialized.

initiator
The system component that originates an I/O command over an I/O bus or network. I/O adapters, network interface cards, and intelligent controller device I/O bus control ASICs are typical initiators. (S) See also *logical unit number*.

input/output (I/O)
Pertaining to a functional unit or communication path involved in an input process, an output process, or both, concurrently or not, and to the data involved in such a process.

instance
An individual object that is a member of some class. In object-oriented programming, an object is created by instantiating a class.

integrity
The ability of a system to either return only correct data or respond that it cannot return correct data.

Internet Protocol (IP)
In the Internet suite of protocols, a connectionless protocol that routes data through a network or interconnected networks and acts as an intermediary between the higher protocol layers and the physical network. IPv4 is the dominant network layer protocol on the Internet, and IPv6 is designated as its successor. IPv6 provides a much larger address space, which enables greater flexibility in assigning addresses and simplifies routing and renumbering.

interswitch link (ISL)
The physical connection that carries a protocol for interconnecting multiple routers and switches in a storage area network.

I/O See *input/output.*

I/O group
A collection of virtual disks (VDisks) and node relationships that present a common interface to host systems.

I/O throttling rate
The maximum rate at which an I/O transaction is accepted for this virtual disk (VDisk).

IP See *Internet Protocol*.

IP address
The unique 32-bit address that specifies the location of each device or workstation in the Internet. For example, 9.67.97.103 is an IP address.

ISL See *interswitch link*.

ISL hop
A hop on an interswitch link (ISL). Considering all pairs of node ports (N-ports) in a fabric and measuring distance only in terms of interswitch links (ISLs) in the fabric, the number of ISLs traversed is the number of ISL hops on the shortest route between the pair of nodes that are farthest apart in the fabric.

J

JBOD (just a bunch of disks)
IBM definition: See *non-RAID*.

HP definition: A group of single-device logical units not configured into any other container type.

L

LBA See *logical block address*.

least recently used (LRU)
An algorithm used to identify and make available the cache space that contains the least-recently used data.

line card
See *blade.*

local fabric
In SAN Volume Controller, those storage area network (SAN) components (such as switches and cables) that connect the components (nodes, hosts, switches) of the local cluster together.

local/remote fabric interconnect
The storage area network (SAN) components that are used to connect the local and remote fabrics together.

logical block address (LBA)
The block number on a disk.

logical unit (LU)
An entity to which Small Computer System Interface (SCSI) commands are addressed, such as a virtual disk (VDisk) or managed disk (MDisk).

logical unit number (LUN)
The SCSI identifier of a logical unit within a target. (S)

longitudinal redundancy check (LRC)
A method of error checking during data transfer that involves checking parity.

LRC See *longitudinal redundancy check*.

LRU See *least recently used*.

LU See *logical unit*.

LUN See *logical unit number*.

LUN masking
A process that allows or prevents I/O to the disk drives through the host-bus-adapter (HBA) device or operating-system device driver.

M

managed disk (MDisk)
A Small Computer System Interface (SCSI) logical unit that a redundant array of independent disks (RAID) controller provides and a cluster manages. The MDisk is not visible to host systems on the storage area network (SAN).

managed disk group
A collection of managed disks (MDisks) that, as a unit, contain all the data for a specified set of virtual disks (VDisks).

Managed Object Format (MOF)
A language for defining Common Information Model (CIM) schemas.

managed space mode
An access mode that enables virtualization functions to be performed. See also *image mode* and *unconfigured mode*.

Management Information Base (MIB)
Simple Network Management Protocol (SNMP) units of managed information that specifically describe an aspect of a system, such as the system name, hardware number, or communications configuration. A collection of related MIB objects is defined as a MIB.

mapping
See *FlashCopy mapping*.

master console
A single point from which to manage the IBM System Storage SAN Volume Controller. For SAN Volume Controller version 4.2.1 and earlier, the master console was purchased either as software that was installed and configured on a server or as a hardware platform with preinstalled operating system and master console software. See *IBM System Storage Productivity Center*.

master virtual disk
The virtual disk (VDisk) that contains a production copy of the data and that an application accesses. See also *auxiliary virtual disk.*

MB See *megabyte*.

MDisk
See *managed disk*.

megabyte (MB)
In decimal notation, 1 048 576 bytes.

mesh configuration
A network that contains a number of small SAN switches configured to create a larger switched network. With this configuration, four or more switches are connected together in a loop with some of the paths short circuiting the loop. An example of this configuration is to have four switches connected together in a loop with ISLs for one of the diagonals.

method
A way to implement a function on a class.

Metro Mirror
A synchronous copy service that enables host data on a particular source virtual disk (VDisk) to be copied to the target VDisk that is designated in the relationship.

MIB See *Management Information Base*.

migration
See *data migration*.

mirrored virtual disk
A virtual disk (VDisk) with two VDisk copies.

mirrorset
IBM definition: See *RAID-1*.

HP definition: A RAID storageset of two or more physical disks that maintain a complete and independent copy of the data from the virtual disk. This type of storageset has the advantage of being highly reliable and extremely tolerant of device failure. Raid level 1 storagesets are referred to as mirrorsets.

MOF See *Managed Object Format (MOF)*.

N

namespace
The scope within which a Common Information Model (CIM) schema applies.

node One SAN Volume Controller. Each node provides virtualization, cache, and Copy Services to the storage area network (SAN).

node name
A name identifier associated with a node. (SNIA)

node port (N_port)
A port that connects a node to a fabric or to another node. N_ports connect to fabric ports (F_ports) or to other N_ports of other nodes. N_ports handle creation, detection, and flow of message units to and from the connected systems. N_ports are end points in point-to-point links.

node rescue
In SAN Volume Controller, the process by which a node that has no valid software installed on its hard disk drive can copy the software from another node connected to the same fibre-channel fabric.

non-RAID
Disks that are not in a redundant array of independent disks (RAID). HP definition: See *JBOD*.

N_port
See *node port*.

O

object In object-oriented design or programming, a concrete realization of a class that consists of data and the operations associated with that data.

object model
A representation, such as a diagram, of objects in a given system. Using symbols similar to standard flowchart symbols, an object model depicts the classes the objects belong to, their associations with each other, the attributes that make them unique, and the operations that the objects can perform and that can be performed on them.

object name
An object that consists of a namespace path and a model path. The namespace path provides access to the Common Information Model (CIM) implementation managed by the CIM Agent, and the model path provides navigation within the implementation.

object path
An object that consists of a namespace path and a model path. The namespace path provides access to the Common Information Model (CIM) implementation managed by the CIM Agent, and the model path provides navigation within the implementation.

offline
Pertaining to the operation of a functional unit or device that is not under the continual control of the system or of a host.

online Pertaining to the operation of a functional unit or device that is under the continual control of the system or of a host.

operating set
In SAN Volume Controller, the set of nodes that are operating together to deliver storage services.

overallocated volume
See *space-efficient virtual disk*.

oversubscription
The ratio of the sum of the traffic that is on the initiator N-node connections to the traffic that is on the most heavily loaded interswitch links (ISLs), where more than one ISL is connected in parallel between these switches. This definition assumes a symmetrical network and a specific workload that is applied equally from all initiators and sent equally to all targets. See also *symmetrical network*.

P

partition
IBM definition: A logical division of storage on a fixed disk.

HP definition: A logical division of a container represented to the host as a logical unit.

partner node
The other node that is in the I/O group to which this node belongs.

partnership
In Metro or Global Mirror operations, the relationship between two

clusters. In a cluster partnership, one cluster is defined as the local cluster and the other cluster as the remote cluster.

paused
In SAN Volume Controller, the process by which the cache component quiesces all ongoing I/O activity below the cache layer.

pend To cause to wait for an event.

petabyte (PB)
In decimal notation, 1 125 899 906 842 624 bytes.

PDU See *power distribution unit*.

physical disk licensing
A type of licensing that grants you the use of a number of physical disks for virtualization. You can also license the use of the Metro Mirror and Global Mirror feature, the use of the FlashCopy feature, or both of these features.

PLUN See *managed disk*.

point-in-time copy
The instantaneous copy that the FlashCopy service makes of the source virtual disk (VDisk). In some contexts, this copy is known as a T_0 *copy*.

port The physical entity within a host, SAN Volume Controller, or disk controller system that performs the data communication (transmitting and receiving) over the fibre channel.

port ID
An identifier associated with a port.

power distribution unit (PDU)
A device that distributes electrical power to multiple devices in the rack. It typically is rack-mounted and provides circuit breakers and transient voltage suppression.

power-on self-test
A diagnostic test that servers or computers run when they are turned on.

prepared
In a Global Mirror relationship, the state that occurs when the mapping is ready to start. While in this state, the target virtual disk (VDisk) is offline.

preparing
In a Global Mirror relationship, the state that occurs when any changed write data for the source virtual disk (VDisk) is flushed from the cache. Any read or write data for the target VDisk is discarded from the cache.

primary virtual disk
In a Metro or Global Mirror relationship, the target of write operations issued by the host application.

property
In the Common Information Model (CIM), an attribute that is used to characterize instances of a class.

PuTTY
A client program that allows you to run remote sessions on your computer through specific network protocols, such as SSH, Telnet, and Rlogin.

Q

qualifier
A value that provides additional information about a class, association, indication, method, method parameter, instance, property, or reference.

quorum
A set of nodes that operates as a cluster. Each node has a connection to every other node in the cluster. If a connection failure causes the cluster to split into two or more groups of nodes that have full connection within the group, the quorum is the group that is selected to operate as the cluster. Typically, this is the larger group of nodes, but the quorum disk serves as a tiebreaker if the groups are the same size.

queue depth
The number of I/O operations that can be run in parallel on a device.

quorum disk
A managed disk (MDisk) that contains a reserved area that is used exclusively for cluster management. The quorum disk is accessed in the event that it is necessary to determine which half of the cluster continues to read and write data.

quorum index
A number that can be either: 0, 1 or 2

R

rack A free-standing framework that holds the devices and card enclosure.

RAID See *redundant array of independent disks*.

RAID 0
IBM definition: RAID 0 allows a number of disk drives to be combined and presented as one large disk. RAID 0 does not provide any data redundancy. If one drive fails, all data is lost.

HP definition: A RAID storageset that stripes data across an array of disk drives. A single logical disk spans multiple physical disks, allowing parallel data processing for increased I/O performance. While the performance characteristics of RAID level 0 is excellent, this RAID level is the only one that does not provide redundancy. Raid level 0 storagesets are referred to as stripesets.

RAID 1
SNIA dictionary definition: A form of storage array in which two or more identical copies of data are maintained on separate media. (S)

IBM definition: A form of storage array in which two or more identical copies of data are maintained on separate media. Also known as mirrorset.

HP definition: See *mirrorset*.

RAID 5
SNIA definition: A form of parity RAID in which the disks operate independently, the data strip size is no smaller than the exported block size, and parity check data is distributed across the array's disks. (S)

IBM definition: See the SNIA definition.

HP definition: A specially developed RAID storageset that stripes data and parity across three or more members in a disk array. A RAIDset combines the best characteristics of RAID level 3 and RAID level 5. A RAIDset is the

best choice for most applications with small to medium I/O requests, unless the application is write intensive. A RAIDset is sometimes called parity RAID. RAID level 3/5 storagesets are referred to as RAIDsets.

RAID 10
A type of RAID that optimizes high performance while maintaining fault tolerance for up to two failed disk drives by striping volume data across several disk drives and mirroring the first set of disk drives on an identical set.

real capacity
The amount of storage that is allocated to a virtual disk copy from a managed disk group.

redundant ac-power switch
A device that provides input power redundancy by attaching a SAN Volume Controller to two independent power sources. If the main source becomes unavailable, the redundant ac-power switch automatically provides power from a secondary (backup) source. When power is restored, the redundant ac-power switch automatically changes back to the main power source.

redundant array of independent disks (RAID)
A collection of two or more disk drives that present the image of a single disk drive to the system. In the event of a single device failure, the data can be read or regenerated from the other disk drives in the array.

redundant SAN
A storage area network (SAN) configuration in which any one single component might fail, but connectivity between the devices within the SAN is maintained, possibly with degraded performance. This configuration is normally achieved by splitting the SAN into two, independent, counterpart SANs. See also *counterpart SAN.*

reference
A pointer to another instance that defines the role and scope of an object in an association.

rejected
A status condition that describes a node that the cluster software has removed from the working set of nodes in the cluster.

relationship
In Metro or Global Mirror, the association between a master virtual disk (VDisk) and an auxiliary VDisk. These VDisks also have the attributes of a primary or secondary VDisk. See also *auxiliary virtual disk, master virtual disk, primary virtual disk, and secondary virtual disk.*

reliability
The ability of a system to continue to return data even if a component fails.

remote fabric
In Global Mirror, the storage area network (SAN) components (switches and cables) that connect the components (nodes, hosts, and switches) of the remote cluster.

roles Authorization is based on roles that map to the administrator and service roles in an installation. The switch translates these roles into SAN Volume Controller administrator and service user IDs when a connection is made to the node for the SAN Volume Controller.

S

SAN See *storage area network*.

SAN Volume Controller fibre-channel port fan in
The number of hosts that can see any one SAN Volume Controller port.

SATA See *Serial Advanced Technology Attachment*.

schema
A group of object classes defined for and applicable to a single namespace. Within the CIM Agent, the supported schemas are the ones that are loaded through the managed object format (MOF).

SCSI See *Small Computer Systems Interface*.

SCSI back-end layer
The layer in a Small Computer Systems Interface (SCSI) network that performs the following functions: controls access to individual disk controller systems that are managed by the cluster; receives requests from the virtualization layer, processes them, and sends them to managed disks; addresses SCSI-3 commands to the disk controller systems on the storage area network (SAN).

SCSI front-end layer
The layer in a Small Computer Systems Interface (SCSI) network that receives I/O commands sent from hosts and provides the SCSI-3 interface to hosts. SCSI logical unit numbers (LUNs) are mapped to virtual disks (VDisks) in this layer as well. Thus, the layer converts SCSI read and write commands that are addressed to LUNs into commands that are addressed to specific VDisks.

SDD See *subsystem device driver (SDD)*.

secondary virtual disk
In Metro or Global Mirror, the virtual disk (VDisk) in a relationship that contains a copy of data written by the host application to the primary VDisk.

Secure Shell (SSH)
A program to log in to another computer over a network, to run commands in a remote machine, and to move files from one machine to another.

Secure Sockets Layer (SSL)
A security protocol that provides communication privacy. With SSL, client/server applications can communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery.

sequential VDisk
A virtual disk that uses extents from a single managed disk.

Serial Advanced Technology Attachment (SATA)
The evolution of the ATA interface from a parallel bus to serial connection architecture. (S)

Serial ATA
See *Serial Advanced Technology Attachment*.

server In a network, the hardware or software that provides facilities to other stations; for example, a file server, a printer server, a mail server. The station making the request of the server is usually called the client.

Service Location Protocol (SLP)
In the Internet suite of protocols, a protocol that identifies and uses network hosts without having to designate a specific network host name.

fibre-channel SFP connector
See *small form-factor pluggable connector*.

Simple Mail Transfer Protocol (SMTP)
An Internet application protocol for transferring mail among users of the Internet. SMTP specifies the mail exchange sequences and message format. It assumes that the Transmission Control Protocol (TCP) is the underlying protocol.

Simple Network Management Protocol (SNMP)
In the Internet suite of protocols, a network management protocol that is used to monitor routers and attached networks. SNMP is an application-layer protocol. Information on devices managed is defined and stored in the application's Management Information Base (MIB).

SLP See *Service Location Protocol*.

Small Computer System Interface (SCSI)
A standard hardware interface that enables a variety of peripheral devices to communicate with one another.

small form-factor pluggable (SFP) connector
A compact optical transceiver that provides the optical interface to a fibre-channel cable.

SMI-S See *Storage Management Initiative Specification*.

SMTP See *Simple Mail Transfer Protocol*.

SNIA See *Storage Networking Industry Association*.

SNMP
See *Simple Network Management Protocol*.

space-efficient VDisk
See *space-efficient virtual disk*.

space-efficient virtual disk
A virtual disk that has different virtual capacities and real capacities.

SSH See *Secure Shell*.

SSPC See *IBM System Storage Productivity Center (SSPC)*.

SSL See *Secure Sockets Layer*.

stand-alone relationship
In FlashCopy, Metro Mirror, and Global Mirror, relationships that do not belong to a consistency group and that have a null consistency group attribute.

stop A configuration command that is used to stop the activity for all copy relationships in a consistency group.

stopped
The status of a pair of virtual disks (VDisks) that have a copy relationship that the user has temporarily broken because of a problem.

storage area network (SAN)
A network whose primary purpose is the transfer of data between computer systems and storage elements and among storage elements. A

SAN consists of a communication infrastructure, which provides physical connections, and a management layer, which organizes the connections, storage elements, and computer systems so that data transfer is secure and robust. (S)

Storage Management Initiative Specification (SMI-S)
A design specification developed by the Storage Networking Industry Association (SNIA) that specifies a secure and reliable interface that allows storage management systems to identify, classify, monitor, and control physical and logical resources in a storage area network. The interface is intended as a solution that integrates the various devices to be managed in a storage area network (SAN) and the tools used to manage them.

Storage Networking Industry Association (SNIA)
An association of producers and consumers of storage networking products whose goal is to further storage networking technology and applications. See www.snia.org.

striped
Pertains to a virtual disk (VDisk) that is created from multiple managed disks (MDisks) that are in the MDisk group. Extents are allocated on the MDisks in the order specified.

stripeset
See *RAID 0*.

subsystem device driver (SDD)
An IBM pseudo device driver designed to support the multipath configuration environments in IBM products.

superuser authority
Can issue any command-line interface (CLI) command. A superuser can view and work with the following panels: View users, Add cluster, Remove cluster, Add users, and Modify users. Only one Superuser role is available.

suspended
The status of a pair of virtual disks (VDisks) that have a copy relationship that has been temporarily broken because of a problem.

switch A network infrastructure component to which multiple nodes attach. Unlike hubs, switches typically have internal bandwidth that is a multiple of link bandwidth, and the ability to rapidly switch node connections from one to another. A typical switch can accommodate several simultaneous full link bandwidth transmissions between different pairs of nodes. (S) Contrast with *hub*.

symmetrical network
A network in which all the initiators are connected at the same level and all the controllers are connected at the same level.

symmetric virtualization
A virtualization technique in which the physical storage in the form of Redundant Array of Independent Disks (RAID) is split into smaller chunks of storage known as *extents*. These extents are then concatenated, using various policies, to make virtual disks (VDisks). See also *asymmetric virtualization*.

synchronized
In Metro or Global Mirror, the status condition that exists when both virtual disks (VDisks) of a pair that has a copy relationship contain the same data.

system
A functional unit, consisting of one or more computers and associated software, that uses common storage for all or part of a program and also for all or part of the data necessary for the execution of the program. A computer system can be a stand-alone unit, or it can consist of multiple connected units.

T

terabyte
In decimal notation, 1 099 511 628 000 bytes.

thinly provisioned volume
See *space-efficient virtual disk*.

topology
The logical layout of the components of a computer system or network and their interconnections. Topology deals with questions of what components are directly connected to other components from the standpoint of being able to communicate. It does not deal with questions of physical location of components or interconnecting cables. (S)

trigger
To initiate or reinitiate copying between a pair of virtual disks (VDisks) that have a copy relationship.

U

UID See *unique identifier*.

unconfigured mode
A mode in which I/O operations cannot be performed. See also *image mode* and *managed space mode*.

uninterruptible power supply
A device that is connected between a computer and its power source that protects the computer against blackouts, brownouts, and power surges. The uninterruptible power supply contains a power sensor to monitor the supply and a battery to provide power until an orderly shutdown of the system can be performed.

unique identifier (UID)
An identifier that is assigned to storage system logical units when they are created. It is used to identify the logical unit regardless of the logical unit number (LUN), status of the logical unit, or whether alternate paths exist to the same device. Typically, a UID is only used once.

unmanaged
An access mode that pertains to a managed disk (MDisk) that is not used by the cluster.

V

valid configuration
A configuration that is supported.

VDisk See *virtual disk (VDisk)*.

VDisk copy
See *virtual disk copy*.

virtual capacity
The amount of storage that is available to a server on a virtual disk (VDisk) copy. In a space-efficient virtual disk, the virtual capacity can be different from the real capacity. In a standard virtual disk, the virtual capacity and real capacity are the same.

virtual disk copy
A physical copy of the data that is stored on a virtual disk (VDisk). Mirrored VDisks have two such copies. Nonmirrored VDisks have one copy.

virtual disk (VDisk)
A device that host systems in a storage area network (SAN) recognize as a Small Computer System Interface (SCSI) disk.

virtualization
In the storage industry, a concept in which a pool of storage is created that contains several disk subsystems. The subsystems can be from various vendors. The pool can be split into virtual disks that are visible to the host systems that use them.

virtualized storage
Physical storage that has virtualization techniques applied to it by a virtualization engine.

virtual storage area network (VSAN)
A fabric within the SAN.

vital product data (VPD)
Information that uniquely defines system, hardware, software, and microcode elements of a processing system.

VLUN See *managed disk*.

VPD See *vital product data*.

VSAN See *virtual storage area network*.

W

WBEM
See *Web-Based Enterprise Management*.

Web-Based Enterprise Management (WBEM)
A tiered, enterprise-management architecture that was developed by the Distributed Management Task Force (DMTF). This architecture provides the management design framework that consists of devices, device providers, the object manager, and the messaging protocol for the communication between client applications and the object manager.

worldwide node name (WWNN)
An identifier for an object that is globally unique. WWNNs are used by Fibre Channel and other standards.

worldwide port name (WWPN)
A unique 64-bit identifier that is associated with a fibre-channel adapter port. The WWPN is assigned in an implementation- and protocol-independent manner.

WWNN
See *worldwide node name*.

WWPN
See *worldwide port name*.

Z

zoning
In fibre-channel environments, the grouping of multiple ports to form a virtual, private, storage network. Ports that are members of a zone can communicate with each other, but are isolated from ports in other zones.

Index

Special characters

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

V

W

Readers' Comments — We'd Like to Hear from You

IBM System Storage SAN Volume Controller
Command-Line Interface User's Guide
Version 4.3.1

Publication No. SC26-7903-04

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Send your comments to the address on the reverse side of this form.

If you would like a response from IBM, please fill in the following information:

Name	Address
Company or Organization	
Phone No.	E-mail address

Readers' Comments — We'd Like to Hear from You
SC26-7903-04

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Information Development
Department 61C
9032 South Rita Road
Tucson, Arizona 85755-4401

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line

Printed in USA

SC26-7903-04

Spine information:

IBM System Storage SAN Volume Controller

SAN Volume Controller Command-Line Interface User's Guide

Version 4.3.1