Dell EMC Storage Systems

Administrator Guide for PowerStore and Unity XT metro node feature

Version 7.0

January 2021

Notes, cautions, and warnings

NOTE: A NOTE indicates important information that helps you make better use of your product.

CAUTION: A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

WARNING: A WARNING indicates a potential for property damage, personal injury, or death.

Contents

1

CLI Workspace and User Accounts

This chapter describes how to use the metro node command line interface (CLI) to configure the CLI workspace and manage user accounts.

Topics:

Configure the CLI workspace

The workspace is the appearance and behavior of a CLI session. Use the procedures described in this section to control the output of commands, the level of logging messages sent to the console, and to search the command history of the current CLI session.

NOTE: Starting the metro node CLI no longer requires a username and password. Please verify that no automated scripts supply usernames or passwords.

Set the threshold for console logging

The console logger displays messages received from directors on the console.

By default, the console displays only emergency (level 0) messages.

Messages are categorized into 8 severities (0-7), with 0 being the most severe:

7 - debug (debug-level messages)

6 - info (informational messages)

5 - notice (normal but significant messages)

4 - warning (warning messages)

3 - err (error messages)

2 - crit (critical messages)

1 - alert (messages that must be handled immediately)

0 - emerg (messages notifying the system as unusable)

To enable messages with lower severity to appear on the console, change the threshold of the logging filter for the console.

1. Use the log filter list command to display existing log filters.

```
VPlexcli:/> log filter list
1. Component='logserver' Destination='null' Consume='true'
2. [Threshold='>0'] Destination='null' Consume='true'
```

2. Determine the ID of the filter controlling the display of messages to the console. The console filter has the following attributes:

   Threhold='>=0'
   Destination= 'null'
   Consume='true'

3. Use the log filter destroy command to delete the existing console logging filter.

```
VPlexcli:> log filter destroy 1
```

4. Use the log filter create command to create a new filter for the console with the required threshold:

```
VPlexcli:> log filter create --threshold <n> --component "logserver"
```

where n is 0-7.

NOTE: The threshold value filters all messages with greater or equal severity.
To see critical (2) and above (0 and 1), set the threshold at 3.
To see error (3) and above (0, 1, and 2) set the threshold at 4.

Set window width to 100

Output from many commands is more than 80 columns wide. Expand the command window in which metro node CLI is running to at least 100 columns in width.

Context tree searching

Search the context tree for context names and data matching specific patterns.

Using the Find command to search the context tree

Use this command to find all contexts matching a pattern. When invoked interactively, the command prints the contexts to the screen.

Patterns can be either literal character strings or strings that include wildcard characters. For a complete list of supported CLI wildcard characters, see the topic "Wildcards" in the *CLI Reference Guide*.

2

Meta Volumes

This chapter describes the procedures to manage metadata and meta-volumes using the metro node CLI.

Topics:

About meta-volumes

metro node metadata includes virtual to physical mappings, data about devices, virtual volumes, and system configuration settings.

Metadata is stored in cache and backed up on specially designated external volumes called meta-volumes.

Meta-volumes are created during system setup.

When a cluster is initially configured, the meta-volume must be the first storage presented to metro node. This prevents the meta-volume from being accidentally overwritten.

After the meta-volume is configured, updates to the metadata are written to both the cache and the meta-volume when the metro node configuration is modified.

Backup meta-volumes are point-in-time snapshots of the current metadata, and provide extra protection before major configuration changes, refreshes, or migrations.

Metadata is read from the meta-volume only during the start of each director.

Meta-volume backups are created:

- Before migrating to a new array
- Before a major update.

Meta-volumes differ from standard storage volumes as mentioned below:

- A meta-volume is created without first being claimed
- Meta-volumes are created directly on storage volumes.

See the *Configuration Guide for metro node* for more details about the criteria to select storage used for meta-volumes.

CAUTION: Do not configure the meta-volume on the vault drives of a storage array. For example, the meta-volume must not be configured on the vault drives of a VNX or CLARiiON array.

Metavolume performance and availability requirements

Performance is not critical for metavolumes. The minimum performance that is allowed is 40 MB/sec and 100 4 K IOP/second.

The physical spindles for metavolumes should be isolated from application workloads.

Dell EMC recommends the following for metavolumes:

- Read caching must be enabled
- A hot spare metavolume must be pre-configured in case of a catastrophic failure of the active metavolume.
- If possible, do not use devices on LUN0. Paths to LUN0 are removed and added whenever its array goes through discovery. This behavior is because LUN0 can be either a default LUN or a real LUN backed by real storage."

Availability is critical for metavolumes. The metavolume is essential for system recovery. The best practice is to mirror the metavolume across two or more back-end arrays to eliminate the possibility of data loss. Choose the arrays that mirror the metavolume so that they are not required to migrate simultaneously.

WARNING: Do not create metavolume using volumes from a single storage array. Single array metavolumes are not a high availability configuration and are a single point of failure.

If metro node temporarily loses access to all metavolumes, the current metadata in cache is automatically written to the metavolumes when access is restored.

If metro node permanently loses access to both metavolumes, it continues to operate based on the metadata in memory. Configuration changes are suspended until a new metavolume is created.

NOTE: If the metro node loses access to all metavolumes, and all directors either fail or restart, changes made to the metadata (the metro node configuration) after access was lost cannot be recovered. System volumes are supported on thinly provisioned LUNs, but these volumes must have thin storage pool resources available, at maximum capacity. System volumes must not compete for this space with user-data volumes in the same pool.

Moving a meta-volume

Steps

1. Use the `ll` command to display a list of storage volumes on the cluster:

```
VPlexcli:/> ll /clusters/cluster-1/storage-elements/storage-volumes
```

```
/clusters/cluster-1/storage-elements/storage-volumes:
Name            VPD83 ID                                 Capacity  Use    Vendor IO    Type
--------------  ---------------------------------------  --------  -------  -----  ------  ----------  Status ----
Clar0068_LUN71  VPD83T3:6006016049e02100281ebe77852cdf11  78G       meta-data  DGC   alive  traditional
Clar0068_LUN74  VPD83T3:6006016049e02100291ebe77852cdf11  78G       meta-data  DGC   alive  traditional
Clar0068_LUN75  VPD83T3:6006016049e02100c064c78a852cdf11  78G       unclaimed  DGC   alive  normal
Clar0068_LUN76  VPD83T3:6006016049e02100c164c78a852cdf11  78G       unclaimed  DGC   alive  normal
.
.
.
```

2. Identify 2 storage volumes that are:
   - Unclaimed
   - 78 GB or larger
   - On different arrays
3. Use the `meta-volume create` command to create a new meta-volume.

   Specify the storage volumes identified in step 2.

```
VPlexcli:/engines/engine-1-1/directors> meta-volume create --name meta_dmx --storage-
volumes VPD83T3:60060160372022009 66da1373865de11,
VPD83T3:60060160372022009 66da1373865de12
```

4. Use the `meta-volume move` command to move the existing in-memory metadata to the new meta-volume:

```
VPlexcli:/engines/engine-1-1/directors> meta-volume move --target-volume meta_dmx
```

Renaming a meta-volume

By default, meta-volume names are based on a timestamp. To change the name, do the following:

Steps

1. Navigate to the `/clusters/cluster/system-volumes/` context:

```
VPlexcli:/> cd clusters/cluster-2/system-volumes/
VPlexcli:/clusters/cluster-2/system-volumes>
```

2. Use the `ll` command to display the names of the meta-volumes.
3. Navigate to the `/clusters/cluster/system-volumes/`*`target-meta-volume`* context.

   For example:

```
VPlexcli:/clusters/cluster-2/system-volumes> cd new_meta1_backup_2010May24_163810
```

4. Use the `set name` *new_meta-volume_name* command to change the name.

   For example:

```
VPlexcli:/clusters/cluster-2/system-volumes/new_meta1_backup_2010May24_163810> set
name backup_May24_pre_refresh
```

Deleting a meta-volume

About this task

NOTE: A meta-volume must be inactive to be deleted. Attempts to delete an active meta-volume fail with an error message.

Steps

1. Navigate to the target volume's context.

   For example:

```
VPlexcli:> cd clusters/cluster-1/system-volumes/metadata_1/
```

2. Use the `ll` command to verify that the volume is not active.

   For example:

```
VPlexcli:/clusters/cluster-1/system-volumes/metadata_1> ll
Attributes:
Name                    Value
----------------------  -----------
active                  false
application-consistent  false
block-count             23592704
block-size              4K
.
.
.
```

3. Use the `meta-volume destroy --meta-volume` *meta-volume* command to delete the specified meta-volume.

   For example:

```
VPlexcli:/clusters/cluster-1/system-volumes/metadata_1> meta-volume destroy --meta-
volume metadata_1
```

A warning message appears:

```
Meta-volume 'metadata_1' will be destroyed. Do you wish to continue? (Yes/No)
```

4. Type `y`.

   NOTE: After the deletion of a meta-data volume, delete the data on the storage volume through external means to avoid any future confusion.

Displaying metavolume

Use the `ll` command to display status for a metavolume:

```
VPlexcli:/clusters/cluster-1/system-volumes/svtmeta> ll
/clusters/cluster-1/system-volumes/svtmeta:
Attributes:
Name                     Value
-----------------------  -----------
active                   true
application-consistent   false
block-count              20971264
block-size               4K
capacity                 80G
component-count          2
free-slots               63997
geometry                 raid-1
health-indications       []
health-state             ok
locality                 local
operational-status       ok
ready                    true
rebuild-allowed          true
rebuild-eta              -
rebuild-progress         -
rebuild-status           done
rebuild-type             full
slots                    64000
stripe-depth             -
system-id                svtmeta
thin-capable             -
transfer-size            128K
volume-type              meta-volume


Contexts:
Name        Description
----------  ------------------------------------------------------------------
components  The list of components that support this device or system virtual
            volume.
```

Use the `ll components/` command to display the component volumes of the metavolume:

```
VPlexcli:/clusters/cluster-2/system-volumes/ICO_META_1_1_Metadata> ll components/
/clusters/cluster-2/system-volumes/clus2_MetaVol/components:
Name                                      Slot    Type            Operational  Health
Capacity
----------------------------------------  Number  --------------  Status       State
--------
----------------------------------------  ------  --------------  -----------  ------
--------
VPD83T3:60000970000192601707533031333136  0       storage-volume  ok           ok
78G
VPD83T3:60060480000190300487533030343445  1       storage-volume  ok           ok
78G
```

Table 1. Metavolume display fields

Field	Description
active	Indicates whether this volume is the currently active metadata volume. The system has only one active metadata volume at a time.
application-consistent	Whether this storage-volume is application-consistent.
block-count	The number of blocks in the volume.
capacity	The size of the metavolume.
component-count	The number of mirrors in the RAID 1 metadata volume.
free-slots	The number of free slots for storage-volume headers in this metavolume.
geometry	Indicates the geometry or redundancy of the device. Always RAID 1.
health-indications	If health-state is not ok, additional information.
health-state	• ok - The storage volume is functioning normally. • degraded - The storage volume may be out-of-date compared to its mirror. (This state applies only to a storage volume that is part of a RAID 1 metadata volume.) • unknown - Metro node cannot determine the health state of the storage volume, or the state is invalid. • non-recoverable error - The storage volume may be out-of-date compared to its mirror (applies only to a storage volume that is part of a RAID 1 metadata volume), and/or metro node cannot determine the health state. • critical failure - Metro node has marked the storage volume as hardware-dead.
locality	Locality of the supporting device. • local - The volume is local to the enclosing cluster. • remote - The volume is made available for a different cluster than the enclosing cluster, and is accessed remotely. • distributed - The virtual volume either has legs, or is capable of having legs at more than one cluster.
operational status	• ok - The storage volume is functioning normally. • degraded - The storage volume may be out-of-date compared to its mirror. (This state applies only to a storage volume that is part of a RAID 1 metadata volume.) • unknown - Metro node cannot determine the health state of the storage volume, or the state is invalid. • error - Metro node has marked the storage volume as hardware-dead. • starting - The storage volume is not yet ready. • lost-communication - The storage volume is unreachable.
ready	Indicates whether this metadata volume is ready or not.
rebuild-allowed	Whether the device is allowed to rebuild.
rebuild-eta	If a rebuild is in progress, the estimated time remaining for the current rebuild to complete.
rebuild-progress	If a rebuild is in progress, the percentage of the device that has been rebuilt.
rebuild-status	The rebuild status of the device. done - Rebuild is complete.
rebuild-type	The rebuild type. • full - A full copy of all the blocks. A metavolume rebuild is always full. • incremental - An incremental copy uses a checksum differencing algorithm to transfer only those blocks that are different. • comparison - A comparison copy. • resync - A resync rewrites blocks affected by a director failure, guaranteeing that the mirror legs are identical.
slots	The total number of slots for storage-volume headers in the metavolume.
stripe-depth	The depth of a stripe in bytes when geometry is RAID-0.

Table 1. Metavolume display fields (continued)

Field	Description
system-id	Name that is assigned to the metavolume.
thin-capable	Indicates if the volume is thin capable. Yes indicates that the volume is thin-capable. - indicates that it is not thin capable.
transfer-size	The transfer size during rebuild in bytes.
volume-type	For metavolumes, it is always meta-volume.

Verifying consistency of a meta-volume

To verify disk consistency of a meta-volume, use the following command:

```
VPlexcli:/>  meta-volume verify-on-disk-consistency -c cluster
```

NOTE: Perform a consistency check on the management server that is local to the cluster you are checking.

3

System Management

This chapter describes how to use the call-home notifications, event log locations, and hardware acceleration with VAAI.

Topics:

Call-home notifications

About call-home notifications

Call-home notifications are messages sent automatically to Dell EMC Customer Service and/or Customer Support Representative when a serious problem occurs. Call-home notifications enable Dell EMC to pro-actively engage the relevant personnel, or use a configured ESRS gateway to resolve the problem.

There are four levels of system events. Call-home notifications are sent only for three levels:

Table 2. Event severity and Call-home notifications

Severity	Definition	Impact on Performance or Availability	Call-home
Critical: (1)	A DU or DL is either highly probable or has occurred.	• System unavailable. • Severe performance degradation.	Yes
Error: (2)	Possible DU or DL. Requires service intervention.	• Limited performance impact. • Loss of redundancy. • Moderate risk of DU/DL.	Yes
Warning: (3)	Service attention required. No urgency.	• No performance impact. • Loss of redundancy. • No risk of data loss or unavailability.	Yes
Info: (4)	Informational event. No action is required.	None.	No

Refer to the SolVe Desktop Troubleshooting Procedures for a list of all events.

Many maintenance activities (such as hardware replacements) generate a flurry of call-home events. Many such procedures include steps to temporarily disable call-home notifications during the operation.

Modify call-home and SYR

Call-home notifications and SYR settings are typically configured during system set-up.

Use the `configuration event-notices-reports-config` CLI command to configure the call-home notifications and/or SYR settings if they were not configured during the initial installation.

The command runs an interview script that prompts for the required information. If either call-home notification or SYR is not configured, interview questions to configure the service that is not configured are displayed.

If both call-home notifications and SYR settings are already configured, the current configuration information is displayed.

Before you begin

You need the following information to complete the configuration of call-home notification:

- IP address of the ESRS gateway used to forward call-home notifications to Dell EMC. Dell EMC recommends using your ESRS gateway as the primary connection address.
- (Optional) One or more IP addresses of secondary ESRS gateway server(s) used to forward call-home notifications to Dell EMC if the primary server fails. These addresses must be different than the address for the primary SESRS gateway server.
- (Optional) One or more e-mail addresses of personnel who should receive e-mail messages when call-home notifications occur.

Additional documentation

See the metro node generator for the procedure to configure SupportAssist.

See the install guide for metro node for information about the supportAssist configuration commands::

- vplex_system_config -support_enable- Enables SupportAssist.
- vplex_system_config -support_disable- Disables SupportAssist.
- vplex_system_config -interview --update-supportassist-gateway - Updates the new gateway information.
- vplex_system_config -reset_supportassist -Removes the SupportAssist configuration.
- vplex_system_config --show-supportassist - Displays existing SupportAssist configuration.

Event log locations

Metro node includes services, processes, components, and operating systems that write entries to various logs.

The system collects logs for:

- Call-home events

The locations of various logs on the metro node management server are listed in the following table:

Table 3. Metro node log file locations

Log name	Description and location
Call home log	On a running management server: • `/opt/dell/vplex/ese/var/log/ESE.log` • `/var/log/VPlex/cli/dreamcatcher.log`
NSFW log	GeoSynchrony log. NSFW sends events to a journald service on the director. The journal service writes NSFW entries to the journal in `/var/log/journal/`. • On a running director: `sudo journalctl -u nsfw` • In collect-diagnostics output: The journal is found `voyager-diagnostics/journal/diagnostic-collection_journal.export`. It requires `systemd-journal-remote` to convert to a journal. 1. `systemd-journal-remote --output=<name>.journal /path/to/journal.export` a. It converts the .export into a file readable by `journalctl`. b. It is required to have a .journal suffix on the output file name. 2. `journalctl --file=<name>.journal <other-flags>` a. It has all the same options available as any other `journalctl` command. 3. `journalctl --file=<name>.journal -u nsfw` a. It limits the journal output to the nsfw unit. It is one example of the many journal flags that one can use.

Hardware acceleration with VAAI

VMware API for Array Integration (VAAI) allows you to:

- Offload storage operations from compute side to storage hardware.
- Shift I/O intensive operations of provisioning and creating a snapshot from hypervisor to metro node.
- Dedicate hypervisor memory and processing resources to other functions.
- UNMAP unused storage blocks from thin provisioned volumes. Thin support in metro node on page 23 Provides more information on thin provisioning.

VAAI is implemented in metro node using four SCSI commands:

- “Compare and Write” offloads coordination of powering virtual machines (VMs) on/off, and moving them between hypervisors.
- “WriteSame (16)” offloads writing same pattern of data, such as zeroing out blocks for disk initialization.
- XCOPY offloads copying data to and from the array through the hypervisor.

  Enabling and disabling XCOPY using CLI provides more information on enabling and disabling XCOPY.

- UNMAP allows the hypervisor to reclaim deleted storage on thin-provisioned metro node virtual storage. See "Understanding Thin Provisioning" for more information on thin-provisioned volume and UNMAP functionality.

Compare and Write

The CompareAndWrite (CAW) SCSI command is used to coordinate VMware operations such as powering-on/off VMs, moving VMs from one ESX to another without halting applications (VMotion), and Distributed Resource Scheduler (DRS) operations.

CAW is used by VMware ESX servers to relieve storage contention, which may be caused by SCSI RESERVATION in distributed virtual machine environments. CAW assists storage hardware acceleration by allowing ESX servers to lock a region of disk instead of an entire disk.

ESX 5.0 servers use this strategy to increase the number of virtual machines that an ESX servers can host, and to increase the performance of those virtual machines.

Support for CAW is enabled by default.

Enabling/disabling CAW

CAUTION: CAW can be enabled/disabled on metro node only by an Dell EMC Customer Support Representative.

VMware servers discover whether the CAW SCSI command is supported:

- During initial storage scanning
- When the VMFS3.HardwareAcceleratedLocking value on the ESX host is enabled (or toggled if it is enabled)

NOTE: To toggle the value: In the vSphere client, toggle host > Configuration > Software > Advanced Settings > VMFS3.HardwareAcceleratedLocking value to 0 and then 1.

If CAW is not supported or support is disabled, metro node returns CHECK CONDITION, ILLEGAL REQUEST, and INVALID OP-CODE. The ESX server reverts to using SCSI RESERVE and the virtual machine operation continues.

Virtual machine operations may experience significant performance degradation if CAW is not enabled.

Metro node enables CAW to be enabled/disabled for all storage associated with metro node, using a single command. When CAW is disabled on metro node, storage volumes, do not include CAW support information in their responses to inquiries from hosts.

To mark storage CAW as disabled:

- VMFS3.HardwareAcceleratedLocking must be toggled, or
- Hosts may need to rescan their storage.

  CAUTION: Enabling/disabling CAW functionality supports exceptional situations such as assisting Dell EMC Technical Support personnel to diagnose a problem. CAW is enabled by default and should be disabled only by Dell EMC Technical Support.

Support for CAW can be enabled or disabled at two levels:

- Storage view - Enabled or disabled for all existing storage views. A storage view created after CAW is enabled/disabled at the storage view level inherits the system default setting. Dell EMC recommends maintaining uniform CAW setting on all storage views. If CAW must be disabled for a given storage view, it must be disabled for all existing and future storage views. To ensure that future storage views to reflect the new setting, change the system default (described below).
- System default - Enabled or disabled as a system default. A storage view created after CAW is enabled/disabled at the system default level inherits the system default setting. If the system default is enabled, CAW support for the new storage view is also enabled.

Display CAW setting

Use the `ls` command in `/clusters/cluster/exports/storage-views` context to display whether CAW is enabled at the storage view level. For example:

```
VPlexcli:/> ll /clusters/cluster-2/exports/storage-views/*
/clusters/cluster-2/exports/storage-views/FE-Logout-test:
Name                        Value
--------------------------  ------------------------------------------------------
caw-enabled                 false
.
.
.
/clusters/cluster-2/exports/storage-views/default_quirk_view:
Name                        Value
--------------------------  ----------------------------------------
caw-enabled                 false
.
.
.
```

Use the `ls` command in `/clusters/cluster` context to display the CAW system default setting:

```
VPlexcli:/> ls /clusters/cluster-1
/clusters/cluster-1:
Attributes:
Name                      Value
------------------------  ----------------------------------------
allow-auto-join           true
auto-expel-count          0
auto-expel-period         0
auto-join-delay           0
cluster-id                1
connected                 true
default-cache-mode        synchronous
default-caw-template      true
.
.
.
```

Enable/disable CAW for a storage view

Use the `set` command in `/clusters/cluster/exports/storage-views/storage-view` context to enable or disable CAW for the storage view.

To enable CAW for a storage view:

```
VPlexcli:/clusters/cluster-1/exports/storage-views/recoverpoint_vols> set caw-enabled
true
```

To disable CAW for a storage view:

```
VPlexcli:/clusters/cluster-1/exports/storage-views/recoverpoint_vols> set caw-enabled
false
```

Enable/disable CAW as system default

Use the `set` command in `/clusters/cluster` context to enable or disable CAW for the entire cluster.

To enable CAW as the cluster system default:

```
VPlexcli:/clusters/cluster-1> set default-caw-template true
```

To disable CAW as the cluster system default:

```
VPlexcli:/clusters/cluster-1> set default-caw-template false
```

CAW statistics

CAW performance statistics are included for front-end volume (fe-lu), front-end port (fe-prt), and front-end director (fe-director) targets.

See Statistics tables on page 97 for a listing of the available statistics. Statistics for fe-director targets are collected as a part of the automatically created perpetual monitor.

You can create a monitor to collect CAW statistics, which can be especially useful for fe-lu targets (because there can be very large numbers of volumes involved, these statistics are not always collected).

WriteSame (16)

The WriteSame (16) SCSI command provides a mechanism to offload initializing virtual disks to metro node. WriteSame (16) requests the server to write blocks of data transferred by the application client to consecutive logical blocks multiple times.

WriteSame (16) is used to offload virtual machine provisioning and snapshotting in vSphere to metro node.

WriteSame (16) enables the array to perform copy operations independently without using host cycles. The array can schedule and execute the copy function much more efficiently.

Metro node support for WriteSame (16) is enabled by default.

Enabling/disabling WriteSame (16)

CAUTION: WriteSame (16) can be enabled/disabled on metro node only by Dell EMC Technical Support personnel.

VMware servers discover whether the WriteSame (16) SCSI command is supported:

- During initial storage scanning
- When the DataMover.HardwareAcceleratedInit value on the ESX host is enabled (or toggled if it is enabled)

NOTE: To toggle the value — In the vSphere client, toggle **host > Configuration > Software > Advanced Settings > DataMover.HardwareAcceleratedInit** value to 0 and then 1.

Virtual Machine operations may experience significant performance degradation if WriteSame (16) is not enabled.

Metro node allows WriteSame (16) to be enabled/disabled for all storage associated with metro node, using a single command. When WriteSame (16) is disabled on metro node, storage volumes, do not include WriteSame (16) support information in their responses to inquiries from hosts.

Support for WriteSame (16) can be enabled or disabled at two levels:

- Storage view - Enabled or disabled for all existing storage views. A storage view created after WriteSame (16) is enabled/disabled at the storage view level inherits the system default setting. Dell EMC recommends maintaining uniform WriteSame (16) setting on all storage views in metro node.

  If WriteSame (16) must be disabled for a given storage view, it must be disabled on all existing and future storage views. To make future storage views to reflect the new setting, change the system default.

- System default - Enabled or disabled as a system default. A storage view created after WriteSame (16) is enabled/disabled at the system default level inherits the system default setting. If the system default is enabled, WriteSame (16) support for the new storage view is also enabled.

CAUTION: To disable the Write Same 16 default template, you must disable Write Same 16 for all existing views, and disable Write Same 16 template so that all future views will be Write Same 16 disabled.
To enable the Write Same 16 default template, you must enable Write Same 16 for all existing views, and enable Write Same 16 template so that all future views will be Write Same 16 enabled.

Display WriteSame (16) setting

Use the `ls` command in `/clusters/cluster/exports/storage-views` context to display whether WriteSame (16) is enabled at the storage view level. For example:

```
VPlexcli:/> ll /clusters/cluster-2/exports/storage-views/*
/clusters/cluster-2/exports/storage-views/FE-Logout-test:
Name                      Value
------------------------  ----------------------------------------
caw-enabled               false
.
.
.
/clusters/cluster-2/exports/storage-views/default_quirk_view:
Name                      Value
------------------------  ----------------------------------------
.
.
.
write-same-16-enabled     false
```

Use the `ls` command in `/clusters/cluster` context to display the WriteSame (16) system default setting:

```
VPlexcli:/> ls /clusters/cluster-1
/clusters/cluster-1:
VPlexcli:/clusters/cluster-1> ls
Attributes:
Name                           Value
-----------------------------  -----------------------------------
allow-auto-join                true
auto-expel-count               0
auto-expel-period              0
auto-join-delay                0
cluster-id                     1
connected                      true
default-cache-mode             synchronous
default-caw-template           true
default-write-same-16-template  false
.
.
.
```

Enable/disable WriteSame (16) for a storage view

Use the `set` command in `/clusters/cluster/exports/storage-views/`*`storage-view`* context to enable or disable WriteSame (16) for the storage view.

To enable WriteSame (16) for a storage view:

```
VPlexcli:/clusters/cluster-1/exports/storage-views/recoverpoint_vols> set write-same-16-enabled true
```

To disable WriteSame (16) for a storage view:

```
VPlexcli:/clusters/cluster-1/exports/storage-views/recoverpoint_vols> set write-same-16-enabled false
```

Enable/disable WriteSame (16) as system default

Use the `set` command in `/clusters/cluster` context to enable or disable WriteSame(16) for the entire cluster.

To enable WriteSame(16) as the cluster system default:

```
VPlexcli:/clusters/cluster-1> set default-write-same-16-template true
```

To disable WriteSame(16) as the cluster system default:

```
VPlexcli:/clusters/cluster-1> set default-write-same-16-template false
```

Offload copy overhead with XCOPY

To minimize I/O overhead and maximize performance on copy operations, data movement should occur as close to the physical storage layer as possible, rather than at the server layer (as in host-based data copies).

Utilizing VMWare's XCOPY feature, metro node manages data allocation and placement using virtual machines, copying data with minimal performance impact on the host. When XCOPY is enabled, on-disk data copy and move operations occur on the storage-array, not on the host.

Enabling and disabling XCOPY using CLI

You can enable or disable XCOPY at the cluster or storage view levels.

XCOPY can be enabled and disabled for all storage views. While it is possible to enable or disable XCOPY for individual views, it is not recommended unless you first consult with Dell EMC Support. The best practice is to always use uniform settings in metro node for all storage views.

1. To enable XCOPY, set the `xcopy-enabled` attribute to true. To disable XCOPY, set the `xcopy-enabled` attribute to false.

   For example, to enable XCOPY for all storage views, enter the following CLI command:

   ```
   VPlexcli:/> set /clusters/**/storage-views/*::xcopy-enabled true
   ```

2. Verify the status of the `xcopy-enabled` attribute by listing all attributes for all storage-views as follows:

   ```
   VPlexcli:/> ll /clusters/cluster-1/exports/storage-views/*
   ```

Enabling and disabling XCOPY by default

XCOPY is enabled by default in metro node because the `xcopy-enabled` attribute is set to true, at manufacturing time, in the cluster context.

To change this behavior, you must alter the default template value of XCOPY.

CAUTION: Changing the default template value of the XCOPY attribute changes the value of the XCOPY attribute in all newly created storage views. This should be done only in rare instances, usually after consultation from Dell EMC Support. Changing the default template value may have an adverse effect on VMWare host I/O performance.

1. **To enable XCOPY by default, set the default-xcopy-template attribute to true as follows:**

   ```
   VPlexcli:/> set /clusters/*::default-xcopy-template true
   ```

2. **Verify the status of the `default-xcopy-template` attribute by listing all attributes of the cluster context as follows:**

   ```
   VPlexcli:/clusters/cluster-1> ls
   ```

Displaying XCOPY statistics

Metro node provides statistics that track performance and frequency of XCOPY operations. These statistics are collected at the front-end.

See Statistics on page 95.

Setting up an XCOPY monitor

For all statistics not automatically collected as a part of perpetual monitoring, you can manually create a monitor to gather statistics of XCOPY latency on a particular metro node virtual volume.

You create a monitor and configure a file sink so that the stats for the particular fe-lu (metro node virtual volume) will be collected in the configured file.

The following example shows how to create a monitor to collect the fe-lu.xcopy-avg-lat statistics for a give volume (VAAI_Vol1_Device_vol) in a file (`/tmp/monitors/director-1-1-A-fe-lu-avg-lat`):

```
VPlexcli:/monitoring/directors/director-1-1-A/monitors> monitor create --name fe-lu-
xcopy-avg-lat
        --director /engines/engine-1-1/directors/director-1-1-A --stats  fe-lu.xcopy-
avg-lat
        --targets /clusters/cluster-1/virtual-volumes/VAAI_Vol1_Device_vol
VPlexcli:/monitoring/directors/director-1-1-A/monitors/fe-lu-ws-avg-lat> monitor add-
file-sink
        /tmp/monitors/director-1-1-A-fe-lu-avg-lat
```

Renaming a metro node cluster

Metro node assigns names to its clusters automatically. By default, clusters are named *cluster-1* and *cluster-2*. You can change these names using the metro node CLI.

After you rename a metro node cluster:

- A running migration job or a scheduled job can fail. To prevent this issue, rename the cluster after the jobs are completed.
- The VPN connectivity can be lost after a system upgrade. Reconfigure VPN after the upgrade.

NOTE: The new name for the cluster can contain up to 63 characters that include uppercase and lowercase letters, numbers and underscores. The name must not start with a number or the prefix `cluster-`. Do not include spaces in the name.

To rename a metro node cluster:

1. Log in to the metro node CLI.
2. Go to the context of the cluster.
3. Type the following command:

   `set name` *name*

   Where *name* is the new name of the cluster.

The following is an example:

```
vplexcli:/clusters/cluster-1>set name clusterone
vplexcli:/clusters/clusterone>
```

LCD Front Panel settings

CAUTION: Do not use the panel to modify any of the settings of iDRAC or R640. Modifying the settings may interfere with metro node settings and result in functionality failure.

4

Thin support in metro node

This chapter describes how metro node supports the thin-aware functionalities.

Topics:

Thin support in metro node

Thin-aware is the functionality of advertising metro node virtual volumes as thin volumes to hosts. Thin volumes offer more efficiency because the amount of resources used is much smaller than allocated. This benefit of providing only the resource needed exceeds the cost of the virtualization technology that is used. It enables dynamic freeing of storage blocks on storage volumes that have thin support. Thin support enables the mapping of one or more logical blocks to physical blocks, when required. The logical blocks provide the storage address space (logical unit capacity) to hosts. Physical storage is only allocated to the logical unit when it is used. This ensures that the logical unit is allocated less physical storage than it reports as its capacity. The physical blocks can be mapped to the logical blocks when required (on write). metro node extends multiple thin capabilities that are provided by the arrays that are attached to the back-end.

Thin storage management

Metro node uses some of the management capabilities of the thin-capable arrays in its back-end to detect and address the storage exhaustion issues. When a host stops using the allocated thin storage blocks from the array, the unused blocks are not freed up and they are not returned to the arrays. For example, in a virtual environment where the datastores of a virtual machine are stored on a thin volume, and these datastores are deleted or moved, the storage space is not freed up. This behavior can result in an out-of-space issue on the thin volumes. When the thin storage capacity reaches a specific threshold, the storage arrays send out events to the hosts indicating that the storage space is diminishing. In such cases, the hosts can send the SCSI UNMAP command to the metro node virtual volumes to free up the unused space.

NOTE: The UNMAP feature is supported only on the thin-enabled metro node virtual volumes that meet the thin requirements. Creating thin-enabled virtual volumes lists the thin requirements for a virtual volume.

Thin rebuild

Metro node provides continuous availability and high availability functionalities through its mirroring capability. During the mirroring process, metro node ensures that a thin mirror leg does not turn into a thick leg. metro node uses its thin rebuild feature to synchronize the data between the mirrors of a RAID-1 device that is built on thin volumes. If the array supports the UNMAP feature, metro node uses the SCSI UNMAP commands to free up space on out of date legs if applicable. If the array does not support the UNMAP feature, metro node writes zeros to blocks that have to be zeroed to preserve thinness. This behavior allows preserving the thinness of the device. Even before the UNMAP support, metro node allowed a metro node administrator to claim a thin storage volume by setting the `thin-rebuild` flag. It directs metro node to make efficient use of the space using thin rebuilds.

Rebuilds of thin provisioned storage provides you more information on the thin provisioned storage rebuilds.

Thin migrations

Metro node supports data mobility features on thin devices. When the migration source or the target is not thin, or the source and the targets are from dissimilar storage-array families, the metro node virtual volume loses its thin properties. In such a case, the virtual volume does not support thin storage management operations. After the migration is completed and committed, the

virtual volume will inherit the thin capabilities of the target device. Migrating thin-capable storage provides you more information on the thin-capable storage migrations.

The following table describes how metro node supports the thin-aware functionalities (based on the understanding of metro node whether the arrays are thin capable).

Table 4. Array thin capability during migration

Functionality	Thin-capable arrays	Arrays that are not thin-capable
Thin provisioning	• Discovers the thin volumes on the backend • Automatically sets the `thin-rebuild` flag as part of the storage volume claiming process • Supports the provisioning of thin volumes on the array through VIAS provisioning • Creates the thin-enabled virtual volumes	• Supports the manual tagging of the thin volumes with the `thin-rebuild` flag as part of the storage volume claiming process
Thin storage management	• Supports the SCSI UNMAP command from the host • Support out-of-space notifications to the host from the last leg that services I/O	Not supported
Thin rebuild	• Automatically sets the `thin-rebuild` flag as part of the storage volume claiming process • Uses the SCSI UNMAP command to free up the storage blocks on the out-of-date leg	• Supports the manual tagging of the thin volumes with the `thin-rebuild` flag as part of the storage volume claiming process • Uses zero writes as part of the mirror synchronization for the unused blocks
Thin migration	• Retains thin storage management capabilities of virtual volume only when migration happens between thin-capable volumes of same storage-array family. • In other scenarios, virtual volume loses thin storage management capabilities during migration, and restores them when migration is committed.	Normal migration behavior with optimization for the unused area.

Thin provisioning

In metro node, thin provisioning is performed through the legacy method (using the EZ provisioning or the Advanced provisioning methods) and through VIAS.

Thin provisioning provides you more information on these methods.

Creating thin-enabled virtual volumes

Metro node supports creating virtual volumes that exhibit thin capabilities to the hosts. To exhibit these capabilities, certain requirements have to be met. The requirements are as follows:

- Storage volumes are provisioned from storage arrays that are supported by metro node as thin-capable (where the thin properties are shown). The storage volumes must also be from a storage-array family that metro node supports (Dell EMC PowerStore, Dell EMC UnityXT). The value corresponding to the `storage-array-family` property must be `XTREMIO`, `CLARiiON`, or `SYMMETRIX` and it must not be `other` or `-`.
- Storage volume display thin properties.
- All the mirrors are created from the same storage-array family that metro node supports (For a RAID-1 configuration). The value corresponding to the `storage-array-family` property must not be `mixed`, `other` or `-`. In the following scenarios, the `thin capable` attribute can show `false` even if the mirrors are created from the same storage-array family that metro node supports:
  - The array software does not support the UNMAP feature
  - The UNMAP feature is not turned on the arrays

Creating thin-enabled virtual volumes through the legacy provisioning method

In the legacy method, you can create a thin-enabled virtual volume in these two ways:

- EZ Provisioning: Use the `storage-tool compose --thin` command to create a virtual-volume on top of the specified storage-volumes, building all intermediate extents, local, and distributed devices as necessary.
- Advanced provisioning: Perform these tasks:
  - Manually claiming thin storage volumes that are discovered by metro node.
  - Creating extents on top of the thin-capable storage volume using the `extent create` command.
  - Creating thin-capable local devices using the `local-device create` command.
  - Creating thin-enabled virtual volumes using the `virtual-volume create --thin` command.

    NOTE: If you create a virtual volume without the `--thin` attribute, a thick volume is created by default. The virtual volume must be built on top of a local RAID 0 device or a RAID 1 device. If you try to create a RAID C local-device with multiple children, or a device that incorporates multiple extents, the created local device is not thin-capable.

The following example shows how to create two extents on top of a thin-capable storage volume (with the restriction that a thick extent is created):

```
VPlexcli:/clusters/cluster-1/storage-elements/storage-volumes> extent create myVolume --
num-extents 2
You are creating 2 extents on top of 1 thin-capable storage-volume 'myVolume'.  The
resulting extents will not be thin-capable.
```

The following example shows how to create an extent that is smaller than the supporting storage volume (with the restriction that a thick extent is created):

```
VPlexcli:/clusters/cluster-1/storage-elements/storage-volumes> extent create myVolume --
size 1MB
The new extent will not completely encompass the following thin-capable storage-volume:
myVolume. The resulting extent will not be thin-capable.
```

Use the following commands to list thin-capable virtual volumes, or to set virtual volumes as thin-enabled:

`virtual-volume list-thin --enabled false --capable true --clusters` *cluster*	List all thin-capable virtual volumes that are not currently thin-enabled.
`virtual-volume list-thin --capable true --clusters` *cluster*	List all thin-capable volumes (whether they are thin-enabled or not).
`virtual-volume set-thin-enabled [true|false] --virtual-volumes` *virtual-volumes*	Set virtual volumes as thin-enabled.

For example, to set all virtual volumes at cluster-1 to thin-enabled, type the following command:

```
virtual-volume set-thin-enabled true --virtual-volumes /clusters/cluster-1/virtual-
volumes/*
```

The *CLI Guide for metro node* provides more information on the commands and their usage.

Changing the thin personality of a virtual volume

Metro node does not report a volume as thin to host initiators until its `thin-enabled` option is set to true (enabled). This value can be set to true as part of the creation process as described in Creating thin-enabled virtual volumes. You can set a virtual volume's `thin-enabled` value to true only if it is thin-capable. Use the `set` command to change the value of the `thin-enabled` attribute to true or false. The value true sets the `thin-enabled` attribute to enabled, and the value false sets the `thin-enabled` attribute as disabled. After the behavior of the virtual volume is changed, the hosts will need to perform certain actions (for example, a rescan) to detect the changed behavior.

```
VPlexcli:/clusters/cluster-2/virtual-volumes/XtremIO_LUN_1_vol> set thin-enabled true

VPlexcli:/clusters/cluster-2/virtual-volumes/XtremIO_LUN_1_vol> ls
Name                          Value
----------------------------  ----------------------------------------
block-count                   5242880
```

```
block-size                    4K
cache-mode                    synchronous
capacity                      20G
consistency-group             -
expandable                    true
expandable-capacity           0B
expansion-method              storage-volume
expansion-status              -
health-indications            []
health-state                  ok
locality                      local
operational-status            ok
scsi-release-delay            0
service-status                running
storage-tier                  -
supporting-device             XtremIO_LUN_1
system-id                     XtremIO_LUN_1_vol
thin-capable                  true
thin-enabled                  enabled
volume-type                   virtual-volume
vpd-id                        VPD83T3:6000144000000010e03e55ee4c98c41f
```

NOTE: You can use wildcards to set multiple metro node virtual volumes to be enabled for thin provisioning, after a metro node software upgrade.

```
/clusters/cluster-1/virtual-volumes/thick_1:
Name                          Value
----------------------------- ----------------------------------------
block-count                                                    52428800
block-size                    4K
cache-mode                    synchronous
capacity                      200G
consistency-group             -
expandable                    true
expandable-capacity           0B
expansion-method              storage-volume
expansion-status              -
health-indications            []
health-state                  ok
locality                      local
operational-status            ok
scsi-release-delay            0
service-status                unexported
storage-tier                  -
supporting-device             device_thick_1_c1
system-id                     thick_1
thin-capable                  false
thin-enabled                  unavailable
volume-type                   virtual-volume
vpd-id                        VPD83T3:6000144000000010e025d83c86ace201
```

Thin storage management

metro node uses some of the management capabilities of the thin-capable arrays in its back-end to detect and address the storage exhaustion issues. It is not mandatory for an array that supports thin volumes to support the thin storage management features. metro node can identify whether an array supports thin storage management features. Based on this discovery, metro node sets the `thin capable` attribute of the virtual volume.

Handling storage exhaustion on thin volumes

A storage array can respond back to metro node with a storage exhaustion error on a write to a thin volume. The storage administrators who continuously monitor storage pool capacity takes necessary actions to avoid any storage block exhaustion in their datacenters.

There are mainly two types of storage block exhaustion errors that a storage array can notify. They are:

- Temporary exhaustion: Occurs when a storage array is in the process of freeing up space and cannot immediately respond back with a success to the write. In such a case, metro node retries I/O for short period of time, before failing the write and marking the storage volume hardware-dead. A call home is issued in such a case and metro node tries to automatically recover the storage volume when it responds successfully to its health tests. If the storage volume is protected by a healthy mirror, the host does not see any disruption to the services as the healthy mirror leg continues to service I/Os to the host.
- Permanent exhaustion: Occurs when there are no more available storage blocks to map to the address to which the host has issued a write command. metro node handles this error differently for mirrored and non-mirrored devices.

For permanent block-resource exhaustion on a non-mirrored storage volume, the requested write is responded with an indication to metro node that the storage volume is write protected because space allocation has failed. metro node virtual volumes also return the same error for the write command back to the host. When VMware hosts receive this error for a write request, they stop the virtual machine that made the write request and allow other virtual machines to continue their operation. Other virtual machines can successfully read and write to the blocks that are already mapped. But, if they make a write request to an unmapped storage block, and that write also gets a resource exhaustion error, they are also stopped.

In a non-mirrored volume, Storage administrators can try reclaiming storage using the UNMAP command and recover from the out of space error condition. If the reclaimed storage is not sufficient, add free block storage to storage arrays to address the space allocation failed error conditions, and then start the virtual machines that are suspended or stopped.

For mirrored volumes, metro node masks the error that occurred on a mirror leg for a host write, like any other I/O error. metro node completes the host request with success when the I/O succeeds on at least one mirror leg. metro node marks the mirror leg Out-Of-Date (OOD) and does not try to rebuild (resurrect) automatically. A storage administrator has to allocate space on the array and to make it available to this storage volume, and then manually recover the mirror leg by following the procedures that are documented in Solve Desktop. Once the mirror has been recovered metro node rebuilds the leg.

If the permanent storage exhaustion occurs on the last leg of a mirrored volume, metro node propagates that error to the host requesting the write as with a non-mirrored volume.

Setting thresholds to the thin storage usage

An administrator can set a soft limit or threshold to certain thinly provisioned storage, which indicates the storage space for the thinly provisioned device is diminishing. This threshold is configured on the host or on the arrays, and not on metro node. The message indicates that the device reached the set threshold. Currently, on receiving such a notification from a storage device, metro node retries the I/O after sending a call home. Such notifications can be received once on an I/O, and the I/O must eventually succeed, unless the thin device runs out of space. On receiving such a call home notification, the metro node administrator can notify the host administrator to either free up space or request the storage administrator to add more capacity.

Thin mirroring and migration

Metro node supports the mirroring of thin volumes and the migration of the thin volumes to different arrays.

During the rebuild of a thin leg, metro node preserves the thin nature of the leg. To do this, metro node issues the SCSI UNMAP command to the arrays that support these commands and writes zeros to the blocks on the arrays that do not support the UNMAP feature. Rebuilds for thin provisioned storage provides you additional information on thin rebuilds.

Performing thin mirroring

If you attach a mirror to a thin-capable device and that mirror is not thin, the resulting RAID 1 device loses its thin-capability.

When you run the `device attach-mirror -d` command to attach a thick mirror leg to a thin-capable device, a warning stating that the device is not thin-capable is displayed. You are also prompted to confirm that you want to continue. You can use the `--force` option to bypass the confirmation, but the resulting device is not thin.

```
VPlexcli:/clusters/cluster-1/storage-elements/extents> device attach-mirror -d myDevice
-m extent_TOP_101_1

The top-level device 'myDevice' is thin-capable. After attaching the mirror, the new
top-level device will not be thin-capable.  Do you wish to proceed? (Yes/No) no

device attach-mirror:  Evaluation of <<device attach-mirror -d myDevice -m
extent_TOP_101_1>> failed.
cause:                 Failed to attach mirror.
```

```
cause:                  Operation was halted by the user
VPlexcli:/clusters/cluster-1/storage-elements/extents>
```

You can attach a mirror to a device already supporting a thin-enabled virtual-volume using the `device attach-mirror` command.

To add a thick mirror leg to a thin-enabled virtual-volume, you can continue by:

- Setting the virtual-volume's thin-enabled property to false using the `set` command. The new virtual-volume is not thin-enabled, nor thin-capable.

```
VPlexcli:/clusters/cluster-1/devices> set ../virtual-volumes/myVolume::thin-enabled
false
VPlexcli:/clusters/cluster-1/devices> device attach-mirror --device myDevice --mirror
myMirror
VPlexcli:/clusters/cluster-1/devices>
```

- Using the `--force` option with the `device attach-mirror` command. The new virtual-volume is not thin-enabled, nor thin-capable.

```
VPlexcli:/clusters/cluster-1/devices> device attach-mirror --device myDevice --mirror
myMirror
VPlexcli:/clusters/cluster-1/devices>
```

In a mirrored thin setup, all the legs must come from the same storage-array family. If you try to create thin legs from the arrays that belong to different storage-array family, the thin personality of the legs are lost and they stop supporting the thin storage management features. The following is an example for such a scenario:

```
VPlexcli:/> device attach-mirror --device xio_device --mirror vnx_device
Thin-capability is only supported with homogeneous storage-array types. The top-level
device
'xio_device' is supported by XtremIO but the mirror 'vnx_device' is supported by
CLARiiON.
Since XtremIO and CLARiiON are not homogeneous, the top-level device will lose thin-
capability
after the new mirror is attached. Do you wish to proceed?  (Yes/No) No

device attach-mirror:  Evaluation of <<device attach-mirror --device xio_device --mirror
vnx_device>>
                       failed.
cause:                 Unable to attach mirror 'vnx_device' to device 'xio_device'.
cause:                 Operation was halted by the user

VPlexcli:/>
```

About thin migrations

Metro node supports the migration of a thin volume to another storage array.

For a thin volume to support the thin storage management features after a migration, both the source and the target volumes must be created from the same storage-array family. If they are created from the arrays that belong to different storage-array family, the `thin-enabled` attribute is retained as **`true`**, the `thin-capable` attribute is set to **`false`** and the UNMAP command is rejected.

Migrating thin capable storage provides you additional information on the thin storage migrations.

5

Provisioning Storage

This chapter describes how to provision storage using metro node integrated storage provisioning.

Topics:

Provisioning Overview

To begin using metro node, you must provision storage so that hosts can access that storage. There are three ways to provision storage in metro node:

- EZ provisioning
- Advanced provisioning

NOTE: Dell EMC recommends using the metro node Unisphere GUI to provision storage.

Provisioning storage using EZ provisioning

EZ provisioning is a simple method of provisioning that is available only in Unisphere for metro node. EZ provisioning creates a virtual volume with a one-to-one mapping to a selected storage volume. Use EZ provisioning to create a virtual volume that uses the entire capacity of the storage volume.

In EZ provisioning, you select storage arrays and define how you want them to be used, protected, and presented to hosts. To provision storage using EZ provisioning, do the following:

1. Register initiators that access metro node storage.
2. Create storage views that include virtual volumes, initiators, and metro node ports to control host access to the virtual volumes.
3. Select the storage array and storage volumes to create virtual volumes.

The Unisphere for metro node Online Help provides more information on provisioning storage using EZ provisioning.

NOTE: In the metro node CLI, you can use the `storage-tool compose` command to create a virtual-volume on top of the specified storage-volumes, building all intermediate extents, local, and distributed devices as necessary. The *CLI Reference Guide for metro node* provides more details on the use of this command.

Changing the thin personality of a virtual volume

Metro node does not report a volume as thin to host initiators until its `thin-enabled` option is set to true (enabled). This value can be set to true as part of the creation process as described in Creating thin-enabled virtual volumes. You can set a virtual volume's `thin-enabled` value to true only if it is thin-capable. Use the `set` command to change the value of the `thin-enabled` attribute to true or false. The value true sets the `thin-enabled` attribute to enabled, and the value false sets the `thin-enabled` attribute as disabled. After the behavior of the virtual volume is changed, the hosts will need to perform certain actions (for example, a rescan) to detect the changed behavior.

```
VPlexcli:/clusters/cluster-2/virtual-volumes/XtremIO_LUN_1_vol> set thin-enabled true

VPlexcli:/clusters/cluster-2/virtual-volumes/XtremIO_LUN_1_vol> ls
Name                          Value
----------------------------  ----------------------------------------
block-count                   5242880
```

```
block-size                    4K
cache-mode                    synchronous
capacity                      20G
consistency-group             -
expandable                    true
expandable-capacity           0B
expansion-method              storage-volume
expansion-status              -
health-indications            []
health-state                  ok
locality                      local
operational-status            ok
scsi-release-delay            0
service-status                running
storage-tier                  -
supporting-device             XtremIO_LUN_1
system-id                     XtremIO_LUN_1_vol
thin-capable                  true
thin-enabled                  enabled
volume-type                   virtual-volume
vpd-id                        VPD83T3:6000144000000010e03e55ee4c98c41f
```

NOTE: You can use wildcards to set multiple metro node virtual volumes to be enabled for thin provisioning, after a metro node software upgrade.

```
/clusters/cluster-1/virtual-volumes/thick_1:
Name                          Value
----------------------------- ----------------------------------------
block-count                                                    52428800
block-size                    4K
cache-mode                    synchronous
capacity                      200G
consistency-group             -
expandable                    true
expandable-capacity           0B
expansion-method              storage-volume
expansion-status              -
health-indications            []
health-state                  ok
locality                      local
operational-status            ok
scsi-release-delay            0
service-status                unexported
storage-tier                  -
supporting-device             device_thick_1_c1
system-id                     thick_1
thin-capable                  false
thin-enabled                  unavailable
volume-type                   virtual-volume
vpd-id                        VPD83T3:6000144000000010e025d83c86ace201
```

6

Volume expansion

This chapter describes how to expand virtual volumes.

Topics:

Overview

A metro node virtual volume is created on a device or a distributed device, and is presented to a host through a storage view. For a number of reasons, you may want to expand the capacity of a virtual volume.

If the volume supports expansion, metro node detects the capacity gained by expansion. Then, you determine the available expansion method: `storage-volume`. Metro node can also detect the available expansion method.

Not all virtual volumes can be expanded. See Determine volume expansion-method for more details.

Perform volume expansion using a simple, non-disruptive procedure:

1. Expand the storage volume associated with the virtual volume on the underlying storage array.
2. Allow metro node to rediscover the underlying storage array.
3. Expand the virtual volume using the CLI or Unisphere.

Additional documentation

- *CLI Guide for metro node*- Run the `virtual-volume expand` command.
- Unisphere for metro node Online Help - Use Unisphere to expand the virtual volume.
- The SolVe Desktop - "Expand a distributed virtual volume using GeoSynchrony" and "Configure storage arrays for metro node."

Volume expansion method

Metro node recommends the best expansion method based on the geometry of the underlying device, using the `expansion-method` attribute.

Possible values for the `expansion-method` attribute are:

- `storage-volume` — Metro node expands the underlying storage volume (the corresponding LUNs on the back-end array).
- `not supported` — Metro node cannot expand the virtual volume because the volume did not meet one or more prerequisites. See Limitations for details.

You can list the `expansion-method` attribute using the CLI or Unisphere.

List expansion-method attribute using CLI

In this example, the expansion-method attribute for Test_volume is displayed by listing the `virtual-volumes` context using the CLI.

```
VPlexcli:> ll /clusters/cluster-1/virtual-volumes/ Test_volume
Name                  Value
```

```
------------------  --------------
.
.
.
capacity              0.5G
consistency-group     -
expandable            true
expandable-capacity   0.0G
expansion-method      storage-volume
expansion-status      -
```

Note that the expansion-method attribute value `storage-volume` indicates that metro node uses the storage volume method to expand this virtual volume by default.

List expansion-method attribute using Unisphere

When using Unisphere, click on the virtual volume name to display the properties of the virtual volume you want to expand.

In the example below, the properties for device_BASIC_vnx-1912_LUN146_1_vol indicate that the recommended expansion method is `storage-volume`. Metro node uses the storage volume method to expand this virtual volume by default.

For more information on using Unisphere to expand a volume, see the Help available on the metro node management server.

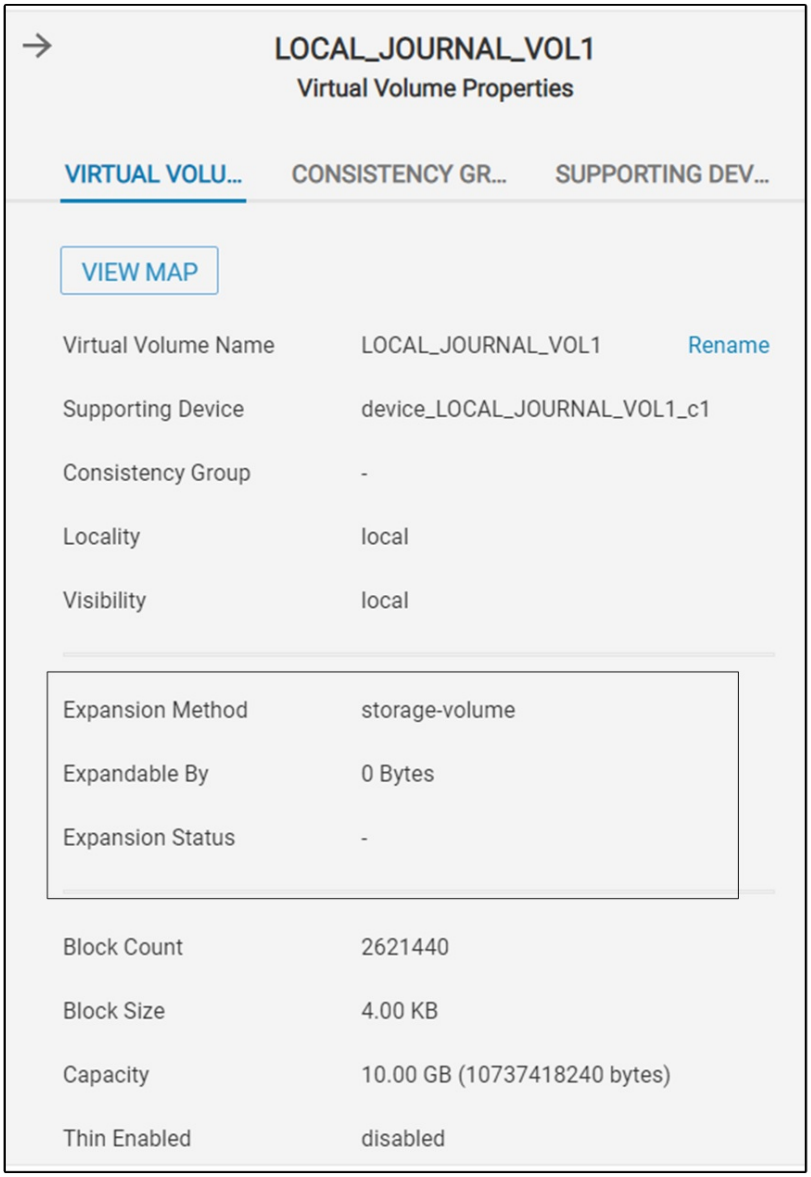

Figure 1. Virtual volume expansion properties (for HTML5)

Expand the virtual volume

Storage-volume expansion method

Use the following guidelines to expand the virtual volume using the storage-volume method.

Overview

The storage volume method of expansion supports simple and fast expansion on a variety of device geometries. Three of the most common device geometries are described here.

1:1 virtual volume to storage volume

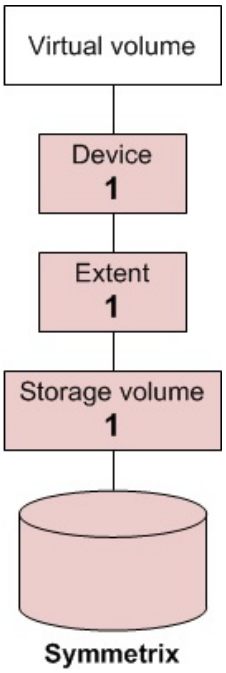

Figure 2. Common geometries: 1:1 virtual volume to storage volume

Dual-legged RAID 1

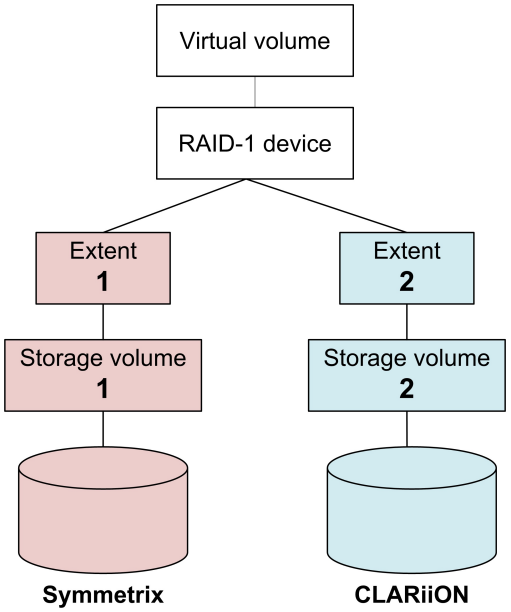

Figure 3. Common geometries: dual-legged RAID 1

Storage-volume expansion method prerequisites

In order to expand a device or add a target for expansion using the storage-volume expansion method, the metro node virtual volume geometry must meet one of the following criteria:

- The virtual volume is mapped 1:1 to the underlying storage volume.
- The virtual volume is a multi-legged RAID 1 volume, and each of its smallest extents is mapped 1:1 to a back end storage volume.
- Volume geometry is a combination of any of the geometries listed previously.

Plan for volume expansion

List the `expandable-capacity` attribute (in the CLI) or the `Expandable By` field (in Unisphere) to plan capacity of your back-end storage devices.

- `expandable-capacity/Expandable By` — For virtual volumes that can be expanded using the storage volume method of expansion, this value is the capacity added to the back-end storage volume, but not yet exposed to the host by the virtual volume.

  This capacity is available for expanding the metro node virtual volume using the storage-volume method of expansion.

  - 0 (zero) - A value of zero indicates that there is no expandable-capacity for the volume. Refer the expansion-method attribute to determine if storage volume based expansion is supported.
  - Non-zero value - A non-zero value indicates capacity available to expand the metro node virtual volume. Examine the `expansion-method` attribute to determine if storage volume based expansion is supported.

Volume Expansion

Perform volume expansion using one of the following techniques:

- The `virtual-volume expand` CLI command. Refer to the *Dell EMC CLI Guide for metro node* for detailed information about this command.
- Expand a virtual volume using the Unisphere. Refer to the Unisphere for metro node Online Help for complete steps.
- Refer to the SolVe Desktop for procedures to expand a distributed virtual volume using GeoSynchrony.

During volume expansion, using the storage-volume method, ensure that:

CAUTION: Performing a major host operation (such as a LIP reset, for example) in order to detect a change in volume size presents risk to volumes accessed by the host. It is best to avoid such resource intensive operations during volume expansion.

- **Expansion initialization traffic occurs on disk areas that are not performing host I/O. In addition, the amount of time taken to initialize the newly added capacity depends on the performance of the array hosting that is the storage volumes. However, the expected performance is still faster than the time taken to rebuild a volume.**
- **Across distributed RAID 1 devices, the initialization process does not consume WAN data bandwidth as each cluster performs its initialization locally.**
- **On RAID 1 and distributed RAID 1 devices, metro node ensures that all RAID 1 legs have consistent information on the expanded space.**
- **The level of redundancy on RAID 1 and distributed RAID 1 device geometries is maintained through the expansion and initialization process.**
- **The newly expanded virtual volume capacity will be available to use by hosts when the initialization process has finished.**
- **If metro node has claimed the storage volumes as thinly provisioned, the initialization process will not affect the underlying provisioning of the additional capacity reported to metro node.**

Check status of volume expansion

Query the status of your volume expansion by listing the value of the following attributes in the `virtual-volumes` context.

- `expansion-status` - Status of virtual volume expansion. Indicates whether a virtual volume expansion is underway or has failed.

  The attribute will have one of the following values:

  - `in-progress` - Expansion is in progress.
  - `failed` - Most recent in-progress expansion has failed and expansion needs to be retried. If expansion is not retried, then this state persists for up to two days. If two days pass with no fix, the failed state clears and the volume is assumed fixed.
  - **`unknown`** - Status cannot be determined. This may be because of a communication error or because of an internal programming error.
  - – (dash character) - None of the above states apply.
- `expansion-summary` - If there are no in-progess or failed expansions, and no virtual volumes with a non-zero expandable-capacity, then the virtual volume summary command displays `No expansion activity` in expansion-summary.

Limitations

The following are general limitations for expanding virtual volumes.

Some virtual volumes cannot be expanded under specific circumstances. Volumes cannot be expanded if any of the following conditions are true:

- Migration or rebuilding is occurring - Expansion is blocked during migration or rebuilds.
  - If you are rebuilding volumes, wait until the rebuild is complete before attempting expansion.
  - If you are migrating data, wait until the migration is complete. Alternatively, cancel or commit the migration, and then perform the expansion.
- Upgrade is occurring - Volume expansion is blocked during Non-Disruptive Upgrade (NDU).
- `health-check` command reports problems - The `health-check` command returns problems relating to the cluster, storage volumes, or virtual volume being expanded.
- Volume is a metadata volume - Metadata volumes cannot be expanded.

Limitations with storage-volume expansion

The following limitations apply to the storage volume expansion method:

- For virtual volumes built on RAID 1 or distributed RAID 1 devices, a maximum of 1000 initialization processes can run concurrently per cluster. If this limit is reached on a cluster, then no new expansions can be started on virtual volumes with these geometries until some of the previously started initialization processes finish on that cluster.

Virtual volumes that do not contain RAID 1 or distributed RAID 1 devices are not affected by this limitation.

Troubleshooting and health indications

When a volume expansion fails, information as to why it failed is added to the `health indications` attribute. When an expansion fails it does not degrade overall health, operational status, or service status of a virtual volume.

The metro node Troubleshooting section of the SolVe Desktop contains procedures for recovering from an error with volume expansions.

Rediscover the array

You may need to rediscover the array after expansion. Depending on the type and configuration of the back-end array, the storage array might not support auto-discovery by metro node.

Best practice

If metro node does not automatically detect the change in the storage volume, use the `array-rediscover` command to force metro node to recognize the back-end expansion.

If you are performing multiple storage volume expansions on the array, complete all storage volume expansions, and rediscover the array only once to force metro node to detect all the expansions.

Some arrays need specific system settings to enable support of auto discovery.

Refer to the SolVe Desktop for the procedures to configure storage arrays for metro node.

NOTE: Review the applicable best practices for host and array connectivity and configuration in the SolVe Desktop. Some arrays require specific settings for automatic detection.

CAUTION: Array rediscoveries may consume excessive resources and can be disruptive to I/O. Rediscover arrays only when necessary.

7

Data migration

This chapter describes data migrations and rebuild.

Topics:

About data migrations

There are two types of data migrations:

- One time migrations - Begin a device migration immediately when the `dm migration start` command is used.
- Batch migrations - Are run as batch jobs using re-usable migration plan files. You can execute multiple device or extent migrations using a single command.

One time migrations

One time migrations includes:

- Device migrations-Devices are 1:1 mapped, or RAID 1 devices built on extents or on other devices.

  Device migrations move data between devices on the same cluster or between devices on different clusters. Use device migrations to:

  - Migrate data between dissimilar arrays.
  - Relocate a hot volume to a faster array.
  - Relocate devices to new arrays in a different cluster.

Limitations

- Device migrations between distributed devices are not supported.
- Devices must be removed from consistency groups before they can be migrated between clusters.

Batch migrations

Batch migrations migrate multiple devices. Create batch migrations to automate routine tasks:

- Use batched device migrations to migrate to dissimilar arrays (you must configure the destination's capacities to match the capacity and tier of the source array), and to migrate devices between clusters in a metro node Metro.

Up to 25 local and 25 distributed migrations can be in progress at the same time. Any migrations beyond those limits are queued until an existing migration completes.

NOTE: Devices must be removed from consistency groups before they can be migrated between clusters.

General procedure to perform data migration

Use the following general steps to perform device migrations:

1. Create and check a migration plan (batch migrations only).
2. Start the migration.
3. Monitor the migration progress.
4. Pause, resume, or cancel the migration (optional).
5. Commit the migration. Commit transfers the source virtual volume, device to the target.

   If the virtual volume on top of a device has a system-assigned default name, committing a device migration renames the virtual volume after the target device.

6. Remove the record of the migration.

Prerequisites for target devices

The target device must:

- Be the same size or larger than the source device.

  If the target is larger in size than the source, the extra space can be utilized by using storage volume expansion, if all the prerequisites for storage volume expansion are met.

  For example, if the source is 200 GB, and the target is 500 GB, only 200 GB of the target can be used after a migration. The remaining 300 GB can be claimed by performing a storage volume expansion, if supported by the virtual volume.

- Not have any existing volumes on it.

  WARNING: Device migrations are not recommended between clusters. All device migrations are synchronous. If there is I/O to the devices being migrated, and latency to the target cluster is equal to or greater than 5ms, then significant performance degradation could occur.

Migrating thin-capable storage

The following table describes the migration scenarios that are supported and the state of the virtual volumes before, during, and after the migration.

Table 5. Migration scenarios

Migration	Virtual volume state before migration	Virtual volume state during migration	Virtual volume state after migration
Thick to thin	Thin-capable = false	Thin-capable = false	Thin-capable = true
	Thin-enabled = unavailable	Thin-enabled = unavailable	Thin-enabled = disabled
	UNMAP rejected	UNMAP rejected	UNMAP rejected **NOTE:** You must set the thin-enabled value to true before UNMAP is processed.
Thin to thin (Thin-enabled virtual volume)	Thin-capable = true	Thin-capable = true	Thin-capable = true
	Thin-enabled = enabled	Thin-enabled = enabled	Thin-enabled = enabled
	UNMAP processed	UNMAP processed	UNMAP processed
Thin to thin (Mixed storage array family)	Thin-capable = true	Thin-capable = false	UNMAP rejected
	Thin-enabled = enabled	Thin-enabled = enabled	UNMAP rejected
Thin to thin (Non thin-enabled virtual volume)	Thin-capable = true	Thin-capable = true	Thin-capable = true
	Thin-enabled = disabled	Thin-enabled = disabled	Thin-enabled = disabled
	UNMAP rejected	UNMAP rejected	UNMAP rejected

Table 5. Migration scenarios (continued)

Migration	Virtual volume state before migration	Virtual volume state during migration	Virtual volume state after migration
			NOTE: In this case, UNMAP is intentionally disabled.
Thin to thick (Thin-enabled virtual volume)	Thin-capable = true	Thin-capable = false	Thin-capable = false
	Thin-enabled = enabled	Thin-enabled = enabled	Thin-enabled = unavailable
	UNMAP processed	UNMAP rejected	UNMAP rejected
Thin to thick (Non thin-enabled virtual volume)	Thin-capable = true	Thin-capable = false	Thin-capable = false
	Thin-enabled = disabled	Thin-enabled = unavailable	Thin-enabled = unavailable
	UNMAP rejected	UNMAP rejected	UNMAP rejected

NOTE:

- During migration, a temporary mirror is created to move data from the migration source to the target. Metro node processes UNMAP commands only when both the `thin-capable` and the `thin-enabled` attributes are set to `true` on the virtual volume.
- If the target of migration is a thin-capable device of larger capacity than the source device, then metro node virtual volumes continue to be thin-capable and preserves the earlier provisioned thin-enabled property after the migration completes. To leverage unused capacity, use the `virtual-volume expand` command.

When migrating from a thin-enabled device to a non-thin device (such as thick device), the volume's thin-enabled attribute stays as `enabled` although UNMAP is rejected during the migration. After the migration completes successfully, the thin-enabled attribute becomes `unavailable` because the target device is thick. This behavior is intended because the volume reverts to a thin volume when migration is aborted or failed."

Consider the following when running one time migrations:

- In a thin to thick extent or device migration (with supported virtual volume), if the source is thin-capable and the target is not thin-capable, the supported virtual volumes are not thin-enabled or thin-capable after migration.

```
VPlexcli:/clusters/cluster-1/devices> dm migration start --paused --name my_migration
--from thin_source
--to device_thick_1
The source 'thin_source' is thin-capable but the target 'device_thick_1' is not thin-
capable. The
virtual-volume 'thin_source_vol' will not be thin-enabled or thin-capable after
migration.
Do you wish to proceed? (Yes/No) no

dm migration start:  Evaluation of <<dm migration start --paused --name my_migration
--from thin_source/
--to device_thick_1_c1/>> failed.
cause:               Failed to create a new data-migration.
cause:               Operation was halted by the user
VPlexcli:/clusters/cluster-1/devices>
```

```
VPlexcli:/clusters/cluster-1/storage-elements/extents> dm migration start --paused --
name my_migration
--from thin_extent_1 --to thick_extent_1
The source 'thin_extent_1' is thin-capable but the target 'thick_extent_1' is not
thin-capable.
The virtual-volume 'thin_source_vol' will not be thin-enabled or thin-capable after
migration.
Do you wish to proceed? (Yes/No) no

dm migration start:  Evaluation of <<dm migration start --paused --name my_migration
--from extent_20
--to extent_31>> failed.
cause:               Failed to create a new data-migration.
```

```
cause:               Operation was halted by the user
VPlexcli:/clusters/cluster-1/storage-elements/extents>
```

- In a thin to thick extent migration (no supported virtual volume), if the source is thin-capable and the target is not thin-capable, the source loses its thin-capability after migration.

```
VPlexcli:/clusters/cluster-1/storage-elements/extents> dm migration start --paused --
name my_migration
--from thin_extent_2 --to thick_extent_1
The source 'thin_extent_2' is thin-capable but the target 'thick_extent_1' is not
thin-capable.
Thin-capability will be lost after migration.  Do you wish to proceed? (Yes/No) no

dm migration start:  Evaluation of <<dm migration start --paused --name my_migration
--from extent_21
--to extent_31>> failed.
cause:               Failed to create a new data-migration.
cause:               Operation was halted by the user
VPlexcli:/clusters/cluster-1/storage-elements/extents>
```

Consider the following when committing one time migrations:

- In a thin to thick device migration, the metro node CLI displays a message stating that the virtual volume's thin-properties are disabled.

```
VPlexcli:/data-migrations/extent-migrations> dm migration commit my_migration --force
The virtual-volume 'my_vol' is no longer thin-capable and will not be thin-enabled
after
migration 'my_migration' is committed.
Committed 1 data migration(s) out of 1 requested migration(s).
VPlexcli:/data-migrations/extent-migrations>
```

- In a thin to thin extent or device migration (with supported virtual-volume), if the thin-enabled value is set to false, there is no change after the migration is committed.

```
VPlexcli:/data-migrations/extent-migrations> dm migration commit my_migration2 --force
Committed 1 data migration(s) out of 1 requested migration(s).
VPlexcli:/data-migrations/extent-migrations>
```

- In a thin to thin device migration (with supported virtual volume), if the thin-enabled value is set to true, the virtual volume remains thin-enabled after the migration is committed.

Consider the following when running and committing batch migrations:

- In a thin to thick device or extent migration, during the check-plan phase, the metro node CLI displays a warning stating that the virtual volumes are not thin-capable or thin-enabled after migration.

```
VPlexcli:/> batch-migrate create-plan --file migration.txt --sources device_thin_1,
device_thin_2
--targets device_thick_1, device_thick_2
Extents matching source pattern: device_thin_1, device_thin_2
Extents matching target pattern: device_thick_2, device_thick_1

Creating file /var/log/VPlex/cli/migration.txt as migration plan file.

Wrote file /var/log/VPlex/cli/migration.txt. Please review and edit this file, and
run this command
in the check-plan phase afterward.

VPlexcli:/> batch-migrate check-plan --file /var/log/VPlex/cli/migration.txt
Checking migration plan file /var/log/VPlex/cli/migration.txt.

WARNING: The source 'device_thin_1' is thin-capable but the target 'device_thick_1'
is not thin-capable.
The virtual-volume 'thin_1' will not be thin-enabled or thin-capable after migration.

WARNING: The source 'device_thin_2' is thin-capable but the target 'device_thick_2'
is not thin-capable.
The virtual-volume 'thin_2' will not be thin-enabled or thin-capable after migration.

Plan-check passed with 2 warnings.
VPlexcli:/>
```

- In a thin to thick extent migration (with no supported virtual-volumes), the metro node CLI displays a warning stating that the source loses its thin-capability after migration.

```
VPlexcli:/> batch-migrate create-plan --file migration.txt --sources extent_thin_1,
extent_thin_2
--targets extent_thick_1, extent_thick_2
Extents matching source pattern: extent_thin_1, extent_thin_2
Extents matching target pattern: extent_thick_2, extent_thick_1

Creating file /var/log/VPlex/cli/migration.txt as migration plan file.

Wrote file /var/log/VPlex/cli/migration.txt. Please review and edit this file, and
run this command
in the check-plan phase afterward.

VPlexcli:/> batch-migrate check-plan --file /var/log/VPlex/cli/migration.txt
Checking migration plan file /var/log/VPlex/cli/migration.txt.

WARNING: The source 'device_thin_1' is thin-capable but the target 'device_thick_1'
is not thin-capable.
Thin-capability will be lost after migration.

WARNING: The source 'device_thin_2' is thin-capable but the target 'device_thick_2'
is not thin-capable.
Thin-capability will be lost after migration.

Plan-check passed with 2 warnings.
VPlexcli:/>
```

- For multiple thin-to-thick migrations, the metro node CLI reports the migration problems with multiple warnings. The following example shows two thin-to-thick migrations, where one migration does not have virtual volumes.

```
VPlexcli:/> batch-migrate check-plan --file /var/log/VPlex/cli/migration.txt
Checking migration plan file /var/log/VPlex/cli/migration.txt.

WARNING: The source 'device_thin_1' is thin-capable but the target 'device_thick_1'
is not thin-capable.
The virtual-volume 'thin_1' will not be thin-enabled or thin-capable after migration.

PROBLEM: Source device '/clusters/cluster-1/devices/device_thin_2' does not have a
volume.

WARNING: The source 'device_thin_2' is thin-capable but the target 'device_thick_2'
is not thin-capable.
Thin-capability will be lost after migration.

Plan-check failed with 1 problem and 2 warnings.
```

- In a simultaneous thin to thick and thick to thin device migration, the virtual volume is not thin-capable or thin-enabled after the batch migration is committed.

Table 6. Simultaneous thin to thick and thick to thin migration

Migration	Source	Target	Volume
BR0_0	device_thick_4	device_thin_4	source_thick
BR0_1	device_thin_5	device_thick_3	source_thin

```
VPlexcli:/> batch-migrate commit --file /var/log/VPlex/cli/migrate.txt
The virtual-volume 'source_thin' is no longer thin-capable and will not be thin-
enabled after
migration 'BR0_1' is committed.
Committed 2 of 2 migrations
VPlexcli:/>
```

About rebuilds

Rebuilds synchronize data from a source drive to a target drive. When differences arise between legs of a RAID, a rebuild updates the out-of-date leg.

There are two types of rebuild behavior:

- A full rebuild copies the entire contents of the source to the target.
- A logging rebuild copies only changed blocks from the source to the target.

Local mirrors are updated using a full rebuild (local devices do not use logging volumes).

In metro node Metro configurations, all distributed devices have an associated logging volume. Logging volumes keep track of blocks that are written during an inter-cluster link outage. After a link or leg is restored, the metro node system uses the information in logging volumes to synchronize mirrors by sending only changed blocks across the link.

Logging volume rebuilds also occur when a leg of a distributed RAID 1 becomes unreachable, but recovers quickly.

If a logging volume is unavailable at the time that a leg is scheduled to be marked out-of-date, the leg is marked as fully out-of-date, causing a full rebuild.

The unavailability of a logging volume matters both at the time of recovery (when the system reads the logging volume) and at the time that a write fails on one leg and succeeds on another (when the system begins writes to the logging volume).

CAUTION: If no logging volume is available, an inter-cluster link restoration causes a full rebuild of every distributed device to which there were writes while the link was down.

See Logging volumes.

Rebuilds for thin provisioned storage

Thin provisioning allows storage to migrate onto a thinly provisioned storage volume while allocating the minimal amount of thin storage pool capacity.

Thinly provisioned storage volumes can be incorporated into RAID 1 mirrors with similar consumption of thin storage pool capacity.

Metro node preserves the unallocated thin pool space of the target storage volume in different ways that are based on whether the target volume is thin-capable or not. For thin-capable volumes, if the source leg indicates zeroed data, metro node issues UNMAP for those blocks on the target volumes. For non thin-capable target legs, metro node checks for zeroed data content before writing, and it suppresses the write where it would cause an unnecessary allocation. For this thin rebuild algorithm to be selected, metro node automatically sets the `thin-rebuild` flag on thin-capable volumes as part of the claiming process. For the storage volumes not supported as thin-capable, the metro node administrator sets a third property, the `thin-rebuild` attribute to `true` during or after the storage claiming.

NOTE: During the storage volume claiming operation, metro node automatically sets the thin rebuild flag to `true` on the thin-capable arrays. Metro node does not perform this activity on the thin storage volumes that are already claimed with the flag set to `false`.

Metro node allows you to change the thin-rebuild value for storage volumes regardless of whether the storage volumes are thin-capable or not. For thin-capable storage volumes, if you try to set the thin-rebuild property to false, the metro node CLI displays a warning. In a scenario where all the contents of the source are written to the target, performance might be better than the normal rebuild if:

- The storage volumes are not thin-capable
- The contents of the source and the target of the rebuild are almost the same
- Only the differing data is written during the thin-rebuild process

The discovered thin provisioning property of storage volumes enables the creation of thin provisioning capable metro node virtual volumes to which hosts can send UNMAP commands to free the unused blocks. However, the configured thin-rebuild property controls the mirror synchronization that is performed at the metro node back-end.

Thin support in metro node provides you more information on the thin-aware capabilities of metro node.

CAUTION: If a thinly provisioned storage volume contains non-zero data before being connected to metro node, the performance of the migration or initial RAID 1 rebuild is adversely affected. If the thin storage allocation pool runs out of space and the leg is the last redundant leg of the RAID 1, further writing to a thinly provisioned device causes the volume to lose access to the device. This issue can cause data unavailability.

Performance considerations

To improve overall metro node performance, disable automatic rebuilds or modify the rebuild transfer size:

- Disable automatic rebuilds to avoid a flood of activity when re-attaching two clusters.

  CAUTION: Disabling automatic rebuilds prevents distributed RAID 1s from synchronizing. Child devices will be out of date, increasing the likelihood of remote reads.

- Modify the rebuild transfer size. For more information, see About transfer-size.

One-time data migrations

A one-time data migration moves data between the specified source and targets as soon as you use the `dm start migration` command. No reusable migration plan file is created as in Batch migrations.

Starting a one-time device migration

Steps

1. Use the `drill down` command to display the components of the source of a view, virtual volume, or device, down to the storage volume level:

   ```
   VPlexcli:/clusters/cluster-1> drill-down -o virtual-volumes/Symm1254_7B7_1_vol
   virtual-volume: Symm1254_7B7_1_vol (cluster-1)
      local-device: Symm1254_7B7_1 (cluster-1)
         extent: extent_Symm1254_7B7_1
            storage-volume: Symm1254_7B7
   ```

2. Identify the device used by the source storage volume.
3. Use the `ll /clusters/cluster-*/devices` command to display available devices.
4. Identify an unused device as the destination.
5. Navigate to the appropriate migration context.

   For device migrations, navigate to `device-migration` context:

   ```
   VPlexcli:/> cd data-migrations/device-migrations
   ```

6. Use the `dm migration start` command to start a migration.

   Specify the --to device by name if that name is unique in the global namespace. Otherwise, specify a full pathname.

   For example:

   ```
   VPlexcli:/data-migrations/device-migrations> dm migration start --name migrate_012 --
   from device_012 --to device_012a --transfer-size 12M
   ```

   CAUTION: Setting a large a transfer size may result in data unavailability. Vary from the default only when performance implications are fully understood.

   If host I/O activity is high, setting a large transfer size may impact host I/O.

   See About transfer-size.

Monitoring a migration's progress

Use the `ls` command to display the migration's status.

About this task

```
VPlexcli:/> ls data-migrations/device-migrations/ migrate_012
Name              Value
----------------  ----------------------------
from-cluster      cluster-1
percentage-done   10
source            device_012
source-exported   false
start-time        Fri May 28 13:32:23 MDT 2010
status            in progress
target            device_012a
target-exported   false
to-cluster        cluster-2
transfer-size     12M
type              full
```

Table 7. Migration status

Field	Description
`from-cluster`	Cluster ID of the source device, or devices in the consistency group.
`percentage-done`	Percentage completion of the migration. 100 % if migration is complete or committed.
`source`	Source device.
`source-exported`	Whether the source device was exported during the migration. Applicable if the migration is an inter-cluster device migration and if the device was not already exported. Devices are exported to a remote cluster to make them visible at that cluster and can be used as a leg in a temporary distributed RAID 1 during the migration. • `false` - Source device was not exported. • `true` - Source device was exported.
`start-time`	Date and time migration was started.
`status`	Status of the migration. • `ready`-The migration is ready. • `queued`-The migration is in the queue. • `in-progress`-The migration is in progress. • `paused`-The migration is paused. • `Commit Pending`-The migration is complete (but not committed). • `committed`-The migration is committed. • `Partially-committed`-The commit operation is failed. • `error`-Error condition, including source or target unreachable. • `cancelled`-The migration is cancelled. • `partially-cancelled` - Attempt to cancel the migration failed. Retry the cancel.
`target`	Destination device.
`target-exported`	Whether the target device was exported during the migration. • `false` - Target device was not exported. • `true` - Target device was exported.
`to-cluster`	Cluster ID of the destination device.
`transfer-size`	Size of the region in cache that is used to service the migration. 40 KB-128 MB.
`type`	Type of rebuild. • `full` - Copies the entire contents of the source to the target. • `logging` - Copies only changed blocks from the source to the target.

Pausing/resuming a migration (optional)

Active migrations (a migration that has been started) can be paused and then resumed at a later time.

About this task

Pause an active migration to release bandwidth for host I/O during periods of peak traffic.

Use the `dm migration pause --migrations` command to pause a migration.

Specify the *migration-name* by name if that name is unique in the global namespace. Otherwise, specify a full pathname.

For example:

- Pause a device migration:

```
VPlexcli:/data-migrations/device-migrations> dm migration pause --migrations
migrate_012
```

Use the `dm migration resume --migrations` command to resume a paused migration.

Specify the *migration-name* by name if that name is unique in the global namespace. Otherwise, specify a full pathname.

For example:

- Resume a paused device migration:

```
VPlexcli:/data-migrations/device-migrations> dm migration resume --migrations
migrate_012
```

Canceling a migration (optional)

Migrations can be canceled in the following circumstances:

About this task

- The migration is in progress or paused. The migration is stopped, and any resources it was using are freed.
- The migration has not been committed. The source and target devices are returned to their pre-migration state.

Use the `dm migration cancel --force --migrations` command to cancel a migration.

Specify the *migration-name* by name if that name is unique in the global namespace. Otherwise, specify a full pathname.

For example:

```
VPlexcli:/data-migrations/device-migrations> dm migration cancel --force --migrations
migrate_012
```

Committing a completed migration

The migration process inserts a temporary RAID 1 structure above the source device with the target as an out-of-date leg of the RAID 1. The migration can be understood as the synchronization of the out-of-date leg (the target).

About this task

After the migration is complete, the commit step detaches the source leg of the RAID 1, and removes the RAID 1.

The virtual volume, or device is identical to the one before the migration except that the source device is replaced with the target device.

A migration must be committed to be cleaned.

CAUTION: Verify that the migration has completed successfully before committing the migration.

Use the dm `migrations commit --force --migrations` *migration-name* command to commit a migration.

NOTE: You must use the `--force` option to commit a migration.

For example:

- Commit a device migration:

```
VPlexcli:/data-migrations/device-migrations> dm migration commit --force --migrations
migrate_012
Committed 1 data migration(s) out of 1 requested migration(s).
```

Cleaning a migration

For *device migrations*, cleaning dismantles the source device down to its storage volumes. The storage volumes that are no longer in use are unclaimed.

For device migrations only, use the `--rename-target` argument to rename the target device after the source device. If the target device is renamed, the virtual volume on top of it is also renamed if the virtual volume has a system-assigned default name.

Without renaming, the target devices retain their target names, which can make the relationship between volume and device less evident.

Use the `dm migration clean --force --migrations` *migration-name* command to clean a migration.

Specify the *migration-name* by name if that name is unique in the global namespace. Otherwise, specify a full pathname.

For example:

```
VPlexcli:/data-migrations/device-migrations> dm migration clean --force --migrations
migrate_012
Cleaned 1 data migration(s) out of 1 requested migration(s).
```

Removing migration records

About this task

NOTE: Migrations must be canceled or committed before they can be removed.

Use the `dm migration remove --force --migrations` *migration-name* command to remove the records of the migration.

Specify the *migration-name* by name if that name is unique in the global namespace. Otherwise, specify a full pathname.

For example:

```
VPlexcli:/data-migrations/device-migrations> dm migration remove --force --migrations
migrate_012
Removed 1 data migration(s) out of 1 requested migration(s).
```

Batch migrations

Batch migrations are run as batch jobs from reusable batch migration plan files. Migration plan files are created using the `create-plan` command.

A single batch migration plan can be for devices.

NOTE: Migrations consume cache resources. Running multiple migrations concurrently may impact host I/O.

Use batch migrations to:

- Retire storage arrays (off-lease arrays) and bring new ones online.
- Migrate devices to a different class of storage array.

The steps to perform a batch migration are generally the same as those described in the General procedure to perform data migration.

There are two additional steps to prepare for a batch migration:

1. Create a batch migration plan file (using the `batch-migrate create-plan` command)
2. Test the batch migration plan file (using the `batch-migrate check-plan` command)

Prerequisites

The following prerequisites are required for batch migrations:

- The source and targets are both devices.
- Local devices must be configured (device migrations) on the target array.
- The structure of the target is the same as the structure of the source.

Creating a batch migration plan

The `batch-migrate create-plan` command creates a migration plan using the specified sources and targets.

About this task

In the following example, the `batch-migrate create-plan` command creates a batch migration named 'MigDev-test.txt' to:

- Migrate two devices at cluster-1 to two devices at cluster-2.
- Overwrite an existing plan with the same name.

```
VPlexcli:/> batch-migrate create-plan --file MigDev-test.txt --sources
/clusters/cluster-1/devices/base0,/clusters/cluster-1/devices/base1 --targets /
clusters/cluster-2/devices/dev1723_618, /clusters/cluster-2/devices/dev1723_61C --
force
Extents matching source pattern: base0, base1
Extents matching target pattern: dev1723_61C, dev1723_618
Creating file /var/log/VPlex/cli/MigDev-test.txt as migration plan file.
Wrote file /var/log/VPlex/cli/MigDev-test.txt. Please review and edit this file, and
run this command in the check-plan phase afterward.
```

In the following example, the `batch-migrate create-plan` command creates a batch migration to migrate all devices at cluster-1 to cluster-2.

```
VPlexcli:/> batch-migrate create-plan migrate.txt --sources /clusters/cluster-1/
devices/* --targets /clusters/cluster-2/devices/*
```

Checking a batch migration plan

The `batch-migrate check-plan` command checks the specified batch migration plan for the following:

About this task

- Device migrations:
  - Target device has no volumes on it
  - Source device has volumes on it

If the migration plan contains errors, a description of the errors is displayed, and the plan check fails. For example:

```
VPlexcli:/> batch-migrate check-plan --file MigDev-test.txt
Checking migration plan file /var/log/VPlex/cli/MigDev-test.txt.
Target device '/clusters/cluster-2/devices/dev1723_61C' has a volume.
Target device '/clusters/cluster-2/devices/dev1723_618' has a volume.
Plan-check failed, 2 problems.
```

Use the steps described in Modifying a batch migration file to correct the plan.

Repeat the process of check and modify until the batch migration plan passes the plan check. For example:

```
VPlexcli:/> batch-migrate check-plan --file migrate.txt
Checking migration plan file /temp/migration_plans/migrate.txt.
Plan-check passed.
```

Modifying a batch migration file

To modify a batch migration file, do one of the following:

About this task

- Use the `batch-migrate create-plan` command, specify the same filename, and use the `--force` option to overwrite the old plan with the new one.
- Exit from the management server, and navigate to `/var/log/VPlex/cli/`.

  Use a text editor (vi) to edit and save the file.

```
VPlexcli:/> exit
Connection closed by foreign host.
service@ManagementServer:~> cd /var/log/VPlex/cli/
service@ManagementServer:/var/log/VPlex/cli>
```

NOTE: To add comments to the migration plan file, add lines beginning with "/ ".

Starting a batch migration

About transfer-size

Transfer-size is the size of the region in cache used to service the migration. The area is globally locked, read at the source, and written at the target.

Transfer-size can be as small 40 K, as large as 128 M, and must be a multiple of 4 K. The default recommended value is 128 K.

A larger transfer-size results in higher performance for the migration, but may negatively impact front-end I/O. This is especially true for metro node Metro migrations.

A smaller transfer-size results in lower performance for the migration, but creates less impact on front-end I/O and response times for hosts.

Set a large transfer-size for migrations when the priority is data protection or migration performance. Set a smaller transfer-size for migrations when the priority is front-end storage response time.

Factors to consider when specifying the transfer-size:

- For metro node Metro configurations with narrow inter-cluster bandwidth, set the transfer size lower so the migration does not impact inter-cluster I/O.
- The region specified by transfer-size is locked during migration. Host I/O to or from that region is held. Set a smaller transfer-size during periods of high host I/O.
- When a region of data is transferred, a broadcast is sent to the system. Smaller transfer-size mean more broadcasts, slowing the migration.

Use the `batch-migrate start` command to start the specified batch migration:

For example:

```
VPlexcli:/> batch-migrate start --file migrate.txt --transfer-size 2M
Started 4 of 4 migrations.
```

Pausing/resuming a batch migration (optional)

Active batch migrations (a migration that has been started) can be paused and resumed.

About this task

Pause an active batch migration to release bandwidth for host I/O during periods of peak traffic.

Resume the batch migration during periods of low I/O.

Use the `batch-migrate pause` command to pause the specified active migration. For example:

```
VPlexcli:/data-migrations/device-migrations> batch-migrate pause --file migrate.txt
```

Use the `batch-migrate resume` command to resume the specified paused migration. For example:

```
VPlexcli:/data-migrations/device-migrations> batch-migrate resume --file migrate.txt
```

Canceling a batch migration (optional)

Cancel an active batch migration to return the source volumes to their state prior to the start of the migration.

About this task

Use the `batch-migrate cancel` command to cancel the specified migration. For example:

```
VPlexcli:/data-migrations/device-migrations> batch-migrate cancel --file migrate.txt
```

NOTE: In order to re-run a canceled migration plan, use the `batch-migrate remove` command to remove the records of the migration. See Removing batch migration records.

Monitoring a batch migration's progress

Use the `batch-migrate summary` command with the `--verbose` option to monitor the progress of the specified batch migration:

About this task

For example:

```
VPlexcli:/data-migrations/device-migrations> batch-migrate summary --file migrate.txt --
verbose
source-                  source-site  target                    target-cluster  migration-
name status                            percentage-complete  eta.
---------------          -----------     ----------------------   --------
  -------------- -------------------  ---
R20061115_Symm2264_010   1               R20070107_Symm2A10_1B0   1
migrate.txt  100                   -
R20061115_Symm2264_011   1               R20070107_Symm2A10_1B1   1
migrate.txt  100                   -
R20061115_Symm2264_012   1               R20070107_Symm2A10_1B2   1
migrate.txt  100                   -
R20061115_Symm2264_0113  1               R20070107_Symm2A10_1B3
1          migrate.txt  27                  4.08min
Processed 4 migrations:
    committed:   0
    complete:    3
    in-progress: 1
    paused:      0
    error:       0
    cancelled:   0
    no-record:   0
```

Viewing a batch migration's status

Use the `batch-migrate summary` command to display the status of the specified batch migration.

About this task

For example:

```
VPlexcli:/> batch-migrate summary migrate.txt
Processed 10 migrations from batch migration BR0:
committed:    0
complete:     10
in-progress:  0
paused:       0
error:        0
cancelled:    0
no-record:    0
```

Table 8. Batch migration summary

Field	Description
`Processed....`	Of the number of source-target pairs specified in the batch migration plan, the number that have been processed.
`committed`	Of the number of source-target pairs that have been processed, the number that have been committed.
`completed`	Of the number of source-target pairs that have been processed, the number that are complete.
`in-progress`	Of the number of source-target pairs that have been processed, the number that are in progress.
`paused`	Of the number of source-target pairs that have been processed, the number that are paused.
`error`	Jobs that encountered errors during processing.
`cancelled`	Of the number of source-target pairs that have been processed, the number that have been cancelled.
`no-record`	Of the number of source-target pairs that have been processed, the number that have no record in the context tree.

NOTE: If more than 25 migrations are active at the same time, they are queued, their status is displayed as `in-progress`, and `percentage-complete` is displayed as "?".

Committing a batch migration

The migration process inserts a temporary RAID 1 structure above the source devices with the target devices as an out-of-date leg of the RAID 1. Migration can be understood as the synchronization of the out-of-date leg (the target).

About this task

After the migration is complete, the commit step detaches the source leg of the RAID 1 and then removes the RAID.

The virtual volume, or device is identical to the one before the migration except that the source device is replaced with the target device.

A migration must be committed in order to be cleaned.

When the batch migration is 100 percent complete, use the `batch-migrate commit` command to replicate the volumes on the target devices and remove the volumes from the source devices.

To commit a batch migration, perform the following:

Steps

1. Use the `batch-migrate summary` command to verify that the migration has completed with no errors.
2. Use the `batch-migrate commit --file` command to commit the migration.

   WARNING: Commit permanently removes the volumes from the source devices.

   For example:

```
VPlexcli:/> batch-migrate commit --file migrate.txt
```

Cleaning a batch migration

For *device migrations*, cleaning dismantles the source device down to its storage volumes. The storage volumes that are no longer in use are unclaimed.

About this task

For device migrations only, use the optional `--rename-target` argument to rename the target device after the source device. If the target device is renamed, the virtual volume on top of it is also renamed if the virtual volume has a system-assigned default name.

Without renaming, the target devices retain their target names, which can make the relationship between volume and device less evident.

Use the `batch-migrate clean --file` command to clean the specified batch migration.

CAUTION: This command must be run before the batch-migration has been removed. The command will not clean migrations that have no record in the VPlexcli context tree.

In the following example, source devices are torn down to their storage volumes and the target devices and volumes are renamed after the source device names

```
VPlexcli:/> batch-migrate clean --rename-targets --file migrate.txt
Using migration plan file /temp/migration_plans/migrate.txt for cleanup phase.
0: Deleted source extent /clusters/cluster-1/devices/R20061115_Symm2264_010, unclaimed
its disks Symm2264_010
1: Deleted source extent /clusters/cluster-1/extents/R20061115_Symm2264_011, unclaimed
its disks Symm2264_011
2: Deleted source extent /clusters/cluster-1/extents/R20061115_Symm2264_012, unclaimed
its disks Symm2264_012
3: Deleted source extent /clusters/cluster-1/extents/R20061115_Symm2264_013, unclaimed
its disks Symm2264_013
```

Removing batch migration records

Remove the migration record only if the migration has been committed or canceled.

About this task

Migration records are in the `/data-migrations/device-migrations` context.

Use the `batch-migrate remove --file` command to remove records for the specified migration.

For example:

```
VPlexcli:/data-migrations/device-migrations> batch-migrate remove --file migrate.txt
```

or:

```
VPlexcli:> batch-migrate remove /data-migrations/device-migrations --file migrate.txt.
```

8

Configure the WAN Network

The two WAN ports on each metro node director support dual 10 Gigabit Ethernet inter-cluster links. The WAN ports are configured as part of the installation of a second cluster. This chapter describes the CLI contexts and procedures to change the configuration created during installation.

Topics:

Metro node hardware and WAN ports

In a metro node Metro over IP cluster, the director has two 10 Gigabit Ethernet (10 GbE) ports named WC-00 and WC-01.

WARNING: Data carried on WAN ports on directors and between clusters in metro node Metro configurations is not encrypted.

To prevent DNS attacks, the WAN ports should be routed only on secure and trusted networks.

Refer to the *Simple Support Matrix for metro node* for information about encryption devices supported in metro node configurations.

Metro over IP WAN port configuration rules

Metro over IP WAN ports must conform to the following rules:

- The two WAN ports on a director should be on different physical networks, and must be on different subnets so that port WC-00 (ip-port-group 0) cannot see port WC-01 (ip-port-group 1) on any director.
- All port WC-00s in the cluster (one from each director) must be in the same subnet and connected to the same LAN. Ports in the same subnet are usually connected to the same Ethernet switch.
- All port WC-01s must be in one subnet, which cannot be the same subnet used for ports WC-00.
- The management port subnet cannot be the same as either of the subnet used for the WAN ports.

Port groups

All ports named WC-00 (in a cluster) are collectively referred to as ip-port-group-0.

All ports named WC-01 (in a cluster) are collectively referred to as ip-port-group-1.

NOTE: Port group names (ip-port-group-0 and ip-port-group-1) cannot be modified.

CLI contexts

The parent context for configuring Ethernet and WAN connections is:

`/clusters/cluster-*/connectivity`

The `/clusters/cluster-*/connectivity` context contains a sub-context for each connectivity role:

- `wan-com` - Configuration of inter-cluster connectivity.
- `local-com` - Configuration of connectivity between local directors.
- `front-end` - Configuration of connectivity with hosts.
- `back-end` - Configuration of connectivity with storage arrays.

port-groups context

The port groups (or communication paths) assigned to each connectivity role (back-end, front-end, local-com or wan-com) are contained in the `port-groups` sub-context of each role.

Ports named WC-00 on each cluster are collectively referred to as ip-port-group-0. There are two ip-port-group-0s, one in each cluster. The ip-port-group-0s on each cluster form one communication channel between the clusters.

Ports named WC-01 on each cluster are collectively referred to as ip-port-group-1. There are two port-group-1s, one in each cluster. The ip-port-group-1s on each cluster form a second communication channel between the clusters.

In the following example, a metro node Metro configuration has two back-end fc-port-groups in each cluster:

```
VPlexcli:/clusters/cluster-1/connectivity/back-end> cd port-groups/
VPlexcli:/clusters/cluster-1/connectivity/back-end/port-groups> ll
Name              Enabled      Member Port
---------------  -----------  -----------
fc-port-group-2  all-enabled  IO-02
fc-port-group-3  all-enabled  IO-03
```

If multiple clusters exist, a local port-group has an analogous port-group of the same name at the remote cluster.

A port-group contains all the ports across all directors that share these characteristics:

- serve the same role
- are the same type
- have the same port number
- are on a slic that is inserted into the same position on the respective directors

Each communication role contains a list of port-groups. Use the `ll` command to display a summary of the role's port-groups:

```
VPlexcli:/clusters/cluster-1/connectivity/wan-com/port-groups> ll
Name              Enabled      Member Port
---------------  -----------  -----------
ip-port-group-0  all-enabled  WC-00
ip-port-group-1  all-enabled  WC-01
```

The Enabled column shows the enabled property of each port-group:

- `all-enabled` - All ports in the port-group are enabled.
- `all-disabled` - All ports in the port-group are disabled.
- inconsistent - All member-ports do not have the same enabled status.

The Member Port column lists the name of the port that belongs to this port-group. If the port name is not the same across all directors, each unique name will be listed.

Use the `set` command on the `enabled` property to modify the enabled status of all reachable ports in the port-group:

- `set enabled all-enabled`: enables all reachable ports in this port-group
- `set enabled all-disabled`: disables all reachable ports in this port-group

port-group subcontexts

Specific `port-group` sub-contexts exist for each port-type: IP (ethernet) and FC (Fibre Channel), if the corresponding ports exist. The sub-contexts associated with a particular port-group depends on both the role the port-group serves and the type of port contained in the port group. A port-group is composed using a port-type prefix and a port-number suffix. Port-type prefixes are:

- FC - Fibre Channel port
- IP - Ethernet port

All port-groups contain a `member-ports` context that provides information about the member-port from each director.

IP port-groups contain:

- `option-set` context contains configuration options common to the member ports.
- `subnet` context contains configuration options for IP networking. Different roles have different networking needs, and thus their subnet contexts contain different properties. These subnets are descried under their associated role.
- `enabled` - Summarizes the enabled status of the individual member ports.

Member ports

All properties under the `member-ports` context are read-only.

All port-groups include a member-ports context that lists the port from each director in the port group. Port-groups will remember member-ports from directors that become unreachable. If a director becomes unreachable, the port-group will display the unreachable ports, but indicate that they are unreachable. Remembering unreachable ports is only possible if the current instance of the CLI learned about the port before the director became unreachable. If a director is unreachable when the CLI starts, its ports will not appear in any port-group.

A long listing of the member-ports context provides a summary of the port-group's member ports:

```
VPlexcli:/clusters/cluster-1/connectivity/wan-com/port-groups/ip-port-group-0/member-
ports> ll
Director         Port   Enabled  Address
---------------  -----  -------  --------------
director-1-1-A   WC-00  enabled  192.168.10.35|
director-1-1-B   WC-00  enabled  192.168.10.36|
```

The `member-ports` context contains a sub-context for each director that contributes a port to the port-group:

```
VPlexcli:/clusters/cluster-1/connectivity/wan-com/port-groups/ip-port-group-0/member-
ports/director-1-1-A> ll
Name       Value
---------  --------------
address    192.168.10.35|
director   director-1-1-A
enabled    enabled
port-name  WC-00
```

These sub-contexts provide limited details about that director's port. You can find full details under the director's `/clusters/*/directors/*/ports` context.

NOTE: The address field is not port-type specific, and will display the address as appropriate for the port type.

subnets context

A subnet is a logical subdivision of an IP network. metro node IP addresses are logically divided into two fields:

- A network or routing prefix.

  On a metro node, the prefix attribute includes a prefix and subnet mask. Specified as an IP address and subnet mask in integer dot notation, separated by a colon.

  For example: 192.168.20.0:255.255.255.0

- A specific identifier for the configuration or the network interface.

  NOTE: Metro node subnet addresses must be consistent, the cluster address and the gateway address must be in the subnet specified by the prefix.

Only ip port-groups have subnet contexts. Use the subnet contexts to display and modify the IP networking configuration used by the member ports. However, since different roles have different networking requirements, the properties in the subnet context are role-dependent.

Subnet attribute requirements:

- `mtu` must be set to a number of bytes between 1024 and 9000.
- `prefix` must contain the IP address of any member ports in the port-group.
- `prefix` must contain the cluster-address.

- `prefix` must contain the gateway.
- `gateway` must be a unique address on the local cluster.

Note the following:

- A cleared address is contained by all prefixes and matches no addresses.
- A cleared prefix contains all addresses.
- A property that is not present in a particular subnet context is considered cleared.

If a change is made to the subnet, the change is validated and applied to all ports using this subnet.

When re-configuring a port-group, there are multiple values that must be consistent with each other. It may be necessary to clear or erase some attribute values before others can be changed.

```
VPlexcli:/clusters/cluster-1/connectivity/wan-com/port-groups/ip-port-group-3> cd
subnets/
VPlexcli:/clusters/cluster-1/connectivity/wan-com/port-groups/ip-port-group-3/subnets> ll
Name
--------------
cluster-1-SN00
cluster-1-SN01
default-subnet
```

To clear a subnet, use `configuration subnet clear` command.

/connectivity/back-end/

The `back-end` role context contains the configuration information required to connect to back-end storage arrays.

The back-end role does not have any associated properties. Note that only IP `port-groups` have subnet contexts.

port-groups/ip-port-group-*/subnet/

The `back-end` role context has subnets that allow you to configure routing to access targets with addresses that are not contained by `prefix`.

The following is a description of the subnet attributes:

- `gateway` - The address of the gateway associated with this subnet.
- `mtu` - The maximum transfer unit for this subnet.
- `prefix` - The prefix and mask for this subnet.
- `remote-subnets` - Prefixes of the remote networks reachable from this subnet.

Refer to `subnets context` for information on modifying or clearing these attributes.

/connectivity/front-end/

The `front-end role` context contains the configuration information required to connect to front-end hosts.

The front-end role has subnets that allow you to configure routing to access hosts with addresses that are not contained by `prefix`. Note that only IP `port-groups` have subnet contexts.

The following is a description of the `/connectivity/front-end/` context subnet attributes:

- `gateway` - The address of the gateway associated with this subnet.
- `mtu` - The maximum transfer unit for this subnet.
- `prefix` - The prefix and mask for this subnet.
- `remote-subnets` - Prefixes of the remote networks reachable from this subnet.

The `remote-subnet` attribute is a list that can be modified using the `configuration subnet remote-subnets add` and `configuration subnet remote-subnets remove` commands.

Refer to `subnets context` for information on modifying or clearing other attributes.

/connectivity/local-com/

The `local` role context contains the configuration information related to inter-director communication within the current cluster.

The local role does not have any associated properties.

Managing and Monitoring back-end network

For high availability each director should have multiple paths to each storage volume. Environmental issues such as network congestion or array problems can affect the availability and performance of these paths. For more details, see the **Best Practices documents for metro node**. Metro node monitors the latency of each back-end IT Nexus, and there is a possibility of encountering poorly performing back-end paths. Metro node has several mechanisms to limit the performance impact:

Taking a Back-End IT Nexus with High Latency Out of Service

If an I/O takes greater than 1 second to complete on an ITL (Initiator-Target-LUN on an IT Nexus), then the ITL and IT accrue a penalty where the allowed command limit for the ITL is reduced from five to one. If the accumulated penalty for an ITL exceeds 2 seconds, then the command limit on the ITL is reduced to zero indicating that no more commands are allowed on this ITL. Due to high latency, if greater than 20 ITLs on an IT Nexus are penalized, then the IT Nexus is marked degraded and metro node automatically stops using the IT Nexus for host-based I/O until performance improves.

NOTE: If the last available path to a Logical Unit is marked degraded, then it cannot be taken out of service, and a penalty is applied to allow a single I/O at a time to the LU. One ITL per Logical Unit per director continues to receive commands. Once the performance improves the metro node automatically restores the default outstanding I/O count to the Logical Unit.

Degraded back-end IT Nexuses can be monitored using the VPlexcli command `back-end degraded list`. For more details, see *CLI Reference Guide for metro node*. Due to continuous high latency, when an IT Nexus is marked degraded, then this command lists the Degradation Reason as **Degraded performance**.

If a user finds that a degraded IT Nexus has been restored to health, it is also possible to manually restore its use through the VPlexcli command `back-end degraded recover`.

Marking a Back-End IT Nexus Isolated due to Unstable Performance

If a back-end IT path is found to cycle between degraded and undegraded three times within a 30 minute period, then the IT Nexus is considered unstable and the metro node automatically stops using the IT Nexus for host-based I/O. In this state, the VPlexcli command `back-end degraded list` list the Degradation Reason as **Isolated due to unstable performance**.

In this case, the IT Nexus remains degraded until the user manually restores it using the VPlexcli command `back-end degraded recover`. It is also possible that threshold reaches to its four-hour default value, after which the IT nexus is marked as **performance degraded** while the recovery process checks its health before undegrading it (and automatically re-enabling the path to serve host-based I/O again if the performance tests pass). If the intermittent latency issue is continued on the IT Nexus and the user is unable to address the root cause, then it is advised to engage metro node Customer Service to manually mark the IT Nexus degraded to remove the path from use until the underlying issue is resolved.

LDAP

The Lightweight Directory Access Protocol (LDAP) is an application protocol for accessing and maintaining distributed directory information services over an Internet Protocol (IP) network. Directory services provides any organized set of records with a hierarchical structure. LDAP is a client-server model protocol.

Directory structure

The organization of a directory is a tree structure. The top most entry in a directory is known as the root entry. This entry normally represents the organization that owns the directory.

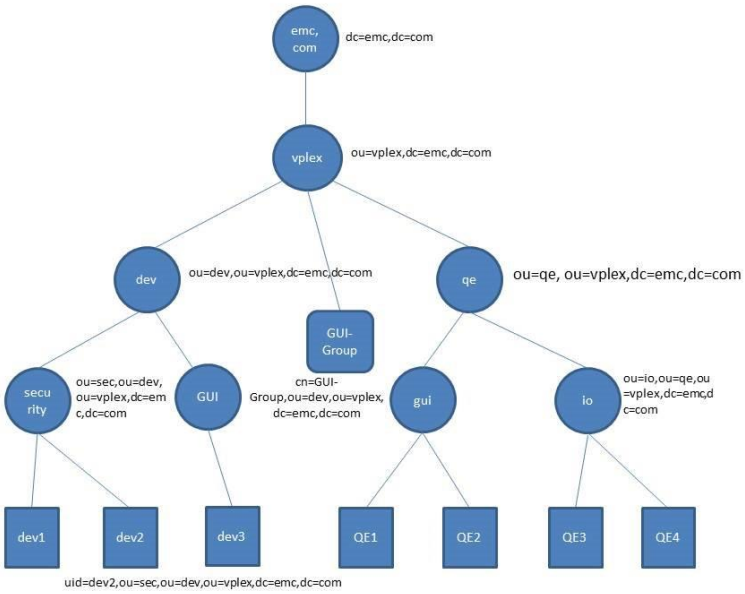

Figure 4. LDAP Directory structure

The metro node SolVe Desktop provides information on configuring LDAP.

Examples (ldapsearch command)

Use the `ldapsearch` command to verify the directory server's attribute mapping values.

- To determine the users that reside under a given organizational unit:

```
service@ManagementServer:~> /usr/bin/ldapsearch -x -LLL -l 30
-H ldap://10.31.50.59:389 -b 'ou=dev,ou=vplex,dc=emc,dc=com' -D
'cn=Administrator,dc=emc,dc=com' objectClass=posixAccount -w password -E pr=1000/
noprompt dn
dn: uid=dev1,ou=security,ou=dev,ou=vplex,dc=emc,dc=com
dn: uid=dev2,ou=security,ou=dev,ou=vplex,dc=emc,dc=com
dn: uid=dev3,ou=GUI,ou=dev,ou=vplex,dc=emc,dc=com
```

- To determine the users that reside under a group principal that must be mapped in the case of Open LDAP servers:

```
service@ManagementServer:~> /usr/bin/ldapsearch -x -LLL -l 30 -
H ldap://10.31.50.59:389 -b 'cn=GUI-Group,ou=vplex,dc=emc,dc=com' -D
'cn=Administrator,dc=emc,dc=com' -w password -E pr=1000/noprompt
dn: cn=GUI-Group,ou=vplex,dc=emc,dc=com
objectClass: groupOfNames
cn: GUI-Group
description: GUI-Group
member: uid=QE1,ou=gui,ou=qe,ou=vplex,dc=emc,dc=com
```

```
member: uid=QE2,ou=gui,ou=qe,ou=vplex,dc=emc,dc=com
member: uid=dev3,ou=GUI,ou=dev,ou=vplex,dc=emc,dc=com
```

- To determine the attributes of the user principal in the case of Open LDAP server:

```
service@ManagementServer:~> /usr/bin/ldapsearch -x -LLL -l 30 -H ldap://
10.31.50.59:389 -b 'uid=dev1,ou=security,ou=dev,ou=vplex,dc=emc,dc=com' -D
'cn=Administrator,dc=emc,dc=com' -w zephyr01 -E pr=1000/noprompt
dn: uid=dev1,ou=security,ou=dev,ou=vplex,dc=emc,dc=com
sn: dev
cn: dev1
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: posixAccount
uid: dev1
loginShell: /bin/bash
homeDirectory: /u/v/x/y/dev1
uidNumber: 50000
gidNumber: 80000
```

9

Consistency Groups

This chapter describes how to manage and operate metro node consistency groups.

Topics:

About metro node consistency groups

Metro node consistency groups aggregate volumes to enable the application of a common set of properties to the entire group.

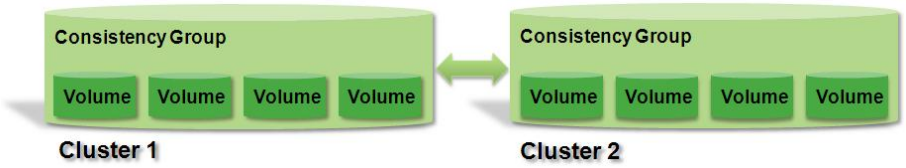

Figure 5. Metro node Consistency group

Synchronous consistency groups

Synchronous consistency groups provide a convenient way to apply rule sets and other properties to a group of volumes in a metro node Local or metro node Metro.

Metro node supports up to 1024 synchronous consistency groups.

A synchronous consistency group:

- Contains up to 1000 virtual volumes.
- Contains either local or distributed volumes, (but not a mixture of both).
- Contains volumes with either global or local visibility.
- Uses write-through caching (known as synchronous cache mode in the metro node user interface).

  Write order fidelity is maintained by completing all writes to disk before acknowledging the write to the host.

The following figure shows a synchronous consistency group that spans two clusters in a metro node Metro configuration.

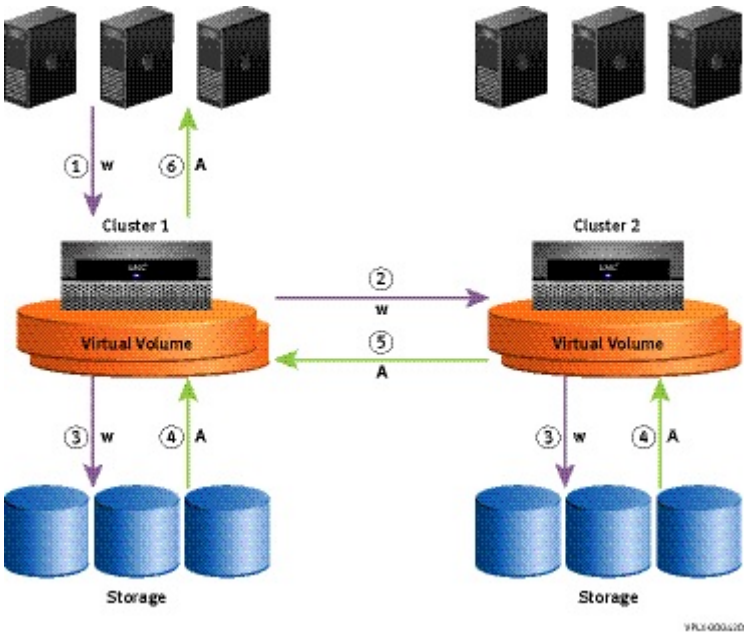

Figure 6. Synchronous consistency group

- The hosts at both clusters write to the metro node distributed volumes in the consistency group.
- Metro node writes data to the back-end storage on both clusters
- An acknowledgment is returned to the host that is issuing the write.

This guarantees that the image on the back end storage is an exact copy on both sides.

Synchronous consistency groups: visibility

Synchronous consistency groups support either distributed or local volumes (but not both in the same consistency group).

Local synchronous consistency groups have only local volumes as members. Local synchronous consistency groups can have the Visibility property set to either:

- Local visibility- The local volumes in the consistency group are visible only to local cluster.
- Global visibility- The local volumes in the consistency group have storage at one cluster, but are visible to both clusters.

Local visibility

Local consistency groups with the Visibility property set only to the local cluster read and write only to their local cluster.

The following figure shows a local consistency group with local visibility.

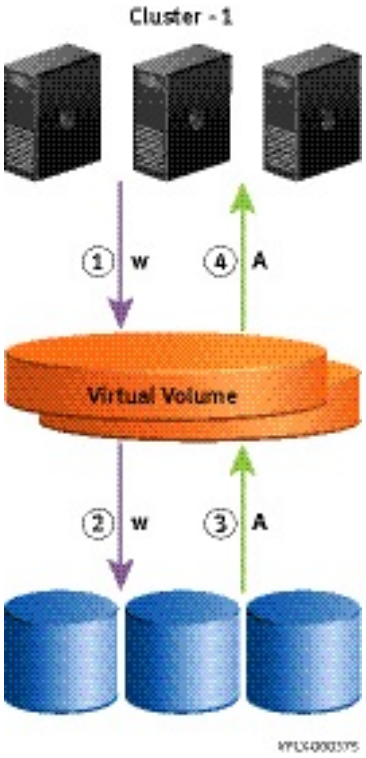

Figure 7. Local consistency groups with local visibility

Global visibility

If the local consistency groups have their Visibility property set to both clusters (global visibility), both clusters can receive I/O from the cluster that does not have a local copy.

All writes from that remote cluster pass over the inter-cluster WAN link before they are acknowledged.

Any reads that cannot be serviced locally are also transferred across the link. This allows the remote cluster to have instant on demand access to the consistency group, but also adds additional latency for the remote cluster.

Local consistency groups with global visibility are supported in metro node Metro environments. Only local volumes can be placed into the local consistency group with global visibility. Local consistency groups with global visibility always use write-through cache mode (synchronous cache mode). I/O that goes to local consistency groups with global visibility will always be synchronous.

the following shows a local consistency group with global visibility.

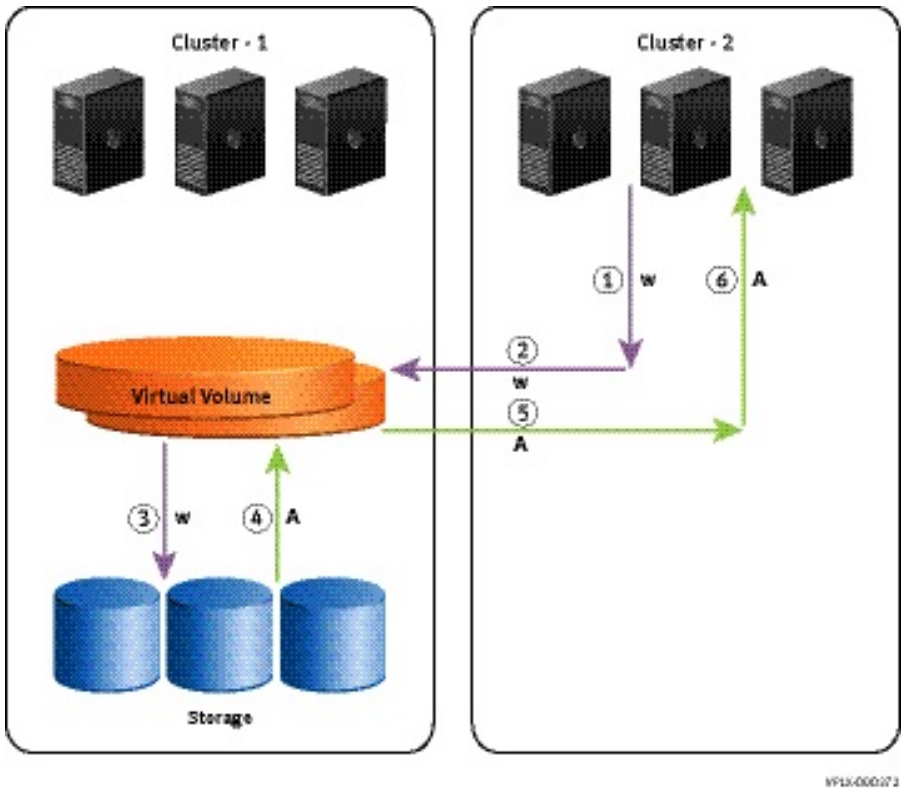

Figure 8. Local consistency group with global visibility

Properties of consistency groups

Properties of a consistency group are applied to all the virtual volumes in the consistency group.

All consistency groups have configurable properties that determine I/O behavior, including:

- Visibility
- Storage-at-clusters
- Detach-rule
- Auto-resume-at-loser
- Virtual-volumes

Visibility

Visibility controls which clusters know about a consistency group.

NOTE: Visibility for consistency groups differs from the visibility property for devices. Devices can have visibility set to `local` (visible only to the local cluster) or `global` (visible to both clusters). All distributed devices have global visibility.

By default, a consistency groups's visibility property is set only to the cluster where the consistency group was created. If a consistency group is created on cluster-2, it is initially visible only on cluster-2.

The visibility of the volumes within the consistency group must match the visibility of the consistency group.

If the visibility of a volume in a consistency group is set to local, the visibility of the consistency group cannot be set to include other clusters. For example, if volume LocalVolume with visibility property set to `local` is added to consistency group TestCG the visibility of TestCG cannot be modified to include other clusters.

In general, visibility is set to one of three options:

- Configure the consistency group to contain only volumes that are local to the local cluster.
- Configure the consistency group to contain only volumes that have storage at one cluster, but have global visibility.
- Configure the consistency group to contain only volumes that are distributed with legs at both clusters.

When a consistency group's visibility is set to a cluster, the consistency group appears below `/clusters/cluster-n/consistency-groups` context for the cluster.

NOTE: The context for a specified consistency group appears in a cluster's consistency group CLI context only if the Visibility property of the consistency group includes that cluster.

Under normal operations, the visibility property can be modified to expand from one cluster to both clusters.

Use the `set` command in `/clusters/cluster/consistency-groups/consistency-group` context to modify the visibility property. If consistency group TestCG is visible only at cluster-1, use the `set` command to make it visible to cluster-1 and cluster-2:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set visibility
cluster-1,cluster-2
```

If a consistency group contains virtual volumes with a given visibility (for example, a member volume's visibility is `local`), the visibility property for the consistency group cannot be changed to conflict with the visibility property of the member virtual volume.

For example, consistency group TestCG is visible only at cluster-1, and contains a volume V whose device is at cluster-1 and has local visibility. Both the following commands will fail, because the volume V is not visible at cluster-2.

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set visibility
cluster-1,cluster-2
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set visibility cluster-2
```

Storage-at-clusters

Storage-at-clusters tells metro node at which cluster the physical storage associated with a consistency group is located.

The `storage-at-clusters` property of a consistency group must be a non-empty subset of the consistency group's `visibility` property.

- If visibility is set to one cluster, then `storage-at-clusters` must be exactly the same as `visibility`.
- If visibility is set to two clusters (1 and 2), then storage-at-clusters can be one of:
  - cluster-1
  - cluster-2
  - cluster-1 and cluster-2

A volume that does not have local storage at every cluster specified by the `storage-at-clusters` property of a consistency group, cannot be added to the consistency group.

For example, if a volume has storage only at cluster-1, it cannot be added to a consistency group that has its `storage-at-cluster` property set to cluster-1 and cluster-2.

A volume that has local storage at more clusters than those specified by the `storage-at-clusters` property of a consistency group, cannot be added to the consistency group.

For example, if a volume has storage at cluster-1 and cluster-2, it cannot be added to a consistency group that has its `storage-at-cluster` property set to cluster-1.

The `storage-at-clusters` property cannot be modified if doing so conflicts with the topology of any of the volumes currently in the consistency group.

Use the `set` command in `/clusters/cluster/consistency-groups/consistency-group` context to modify the `storage-at-clusters` property. For example, to set the `storage-at-clusters` property to both clusters:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set storage-at-clusters
cluster-1,cluster-2
```

NOTE: Best practice is to set the `storage-at-clusters` property when the consistency group is empty.

Detach-rule

Detach rules are a consistency group's policy for automatically picking a *winning* cluster when there is an inter-cluster link outage.

For metro node Metro configurations, there are two consistency group detach rules:

- `no-automatic-winner` - The consistency group does not select a winning cluster.
- `winner cluster-name delay seconds` - The cluster specified by cluster-name is declared the winner if an inter-cluster link outage lasts more than the number of seconds specified by delay.

If a consistency group has a detach rule configured, the rule applies to all volumes in the consistency group and overrides any rule-sets applied to individual volumes.

This property is not applicable for local consistency groups.

By default, no specific detach rule is configured for a consistency group. Instead, the `no-automatic-winner` detach rule is set as default for a consistency group with visibility to both clusters.

Best practice is to apply detach rules to a consistency group that meet the needs of your application in terms of I/O continuance and data loss tolerance.

Use the `consistency-group set-detach-rule` commands to configure the detach-rule for a consistency group:

- Use the `consistency-group set-detach-rule no-automatic-winner` command to set the detach-rule as `no-automatic-winner`:

  ```
  VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set-detach-rule no-automatic-winner
  ```

- Use the `consistency-group set-detach-rule winner` command to specify which cluster is the winner, and the number of seconds metro node waits after a link outage before detaching the winning cluster:

  ```
  VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set-detach-rule winner --cluster cluster-1 --delay 5s
  ```

The following table describes the detach rule behavior for synchronous consistency group.

Table 9. Detach rule behavior - synchronous consistency group

Detach rule	Behavior (irrespective of the cluster on which I/O is happening)
Cluster-1 wins	I/O is allowed on Cluster-1
	I/O suspends at Cluster-2
	No data loss / no data rollback
Cluster-2 wins	I/O suspends at Cluster-1
	I/O is allowed on Cluster-2
	No data loss / no data rollback
No automatic winner[a]	I/O suspends at Cluster-1
	I/O suspends at Cluster-2
	No data loss / no data rollback

a. DU on both clusters if WAN-COM connectivity between metro node clusters goes down

Note the following:

- Active I/O indicates active writes.
- The detach rule behavior described the previous table are based on the assumption that there is a healthy leg in the winner cluster, during the time of cluster partition.
- Use the `consistency-group resume-after-rollback` command to resume after roll back.
- In case of detach rule `no-automatic-winner`, to resume I/O, you must manually choose a cluster as the winner. Use the `consistency-group choose-winner` command to choose a winner.

Auto-resume-at-loser

Determines whether the loser automatically resumes I/O when the inter-cluster link is repaired after a failure.

When the link is restored, the losing cluster finds out that the data on the winning cluster is different. The loser must determine whether to suddenly change to the winner's data, or to keep suspending I/O.

By default, `auto-resume` is enabled.

Usually, this property is set to `false` to give the administrator time to halt and restart the application. Otherwise, dirty data in the host's cache may be inconsistent with the image on disk to which the winning cluster has been writing. If the host flushes dirty pages out of sequence, the data image may be corrupted.

Set this property to `true` for consistency groups used in a cluster cross-connect. In this case, there is no risk of data loss since the winner is always connected to the host, avoiding out of sequence delivery.

`true` (default) - I/O automatically resumes on the losing cluster after the inter-cluster link has been restored.

Set `auto-resume-at-loser` to `true` only when the losing cluster is servicing a read-only application such as servicing web pages.

`false` - I/O remains suspended on the losing cluster after the inter-cluster link has been restored. I/O must be manually resumed.

Set `auto-resume-at-loser` to `false` for all applications that cannot tolerate a sudden change in data.

CAUTION: Setting auto-resume property to `true` may cause a spontaneous change of the data view presented to applications at the losing cluster when the inter-cluster link is restored. If the application has not failed, it may not be able to tolerate the sudden change in the data view and this can cause data corruption. Set the property to `false` except for applications that can tolerate this issue and for cross connected hosts.

Use the `set` command in the advanced context to configure the auto-resume property for a consistency group:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG/advanced> set auto-resume-at-
loser true
```

Virtual-volumes

Administrators can add and remove virtual volumes to a consistency group. In order to be added to a consistency group, a virtual volume:

- Must not be a logging volume
- Must have storage at every cluster in the `storage-at-clusters` property of the target consistency group
- Must not be a member of any other consistency group
- Any properties (such as detach rules or auto-resume) that conflict with those of the consistency group are automatically changed to match those of the consistency group

NOTE: Virtual volumes with different properties are allowed to join a consistency group, but inherit the properties of the consistency group.

Use the `consistency-group list-eligible-virtual-volumes` command to display virtual volumes that are eligible to be added to a consistency group.

Use the `consistency-group add-virtual-volumes` command to add one or more virtual volumes to a consistency group.

Use the `ll /clusters/cluster-*/consistency-groups/`*consistency-group* command to display the virtual volumes in the specified consistency group.

Use the `consistency-group remove-virtual-volumes` command to remove one or more virtual volumes from a consistency group.

Manage consistency groups

NOTE: A key best practice for creating and managing consistency groups is to create a 1:1 relationship between consistency groups and applications. All volumes (and only those volumes) required for an application should be in a single consistency group.

Creating a consistency group

Before you create a consistency group, consider its usage:

About this task

- At which clusters is the underlying storage of the virtual volumes located? If volumes are at both clusters, set the storage-at-cluster property as `cluster-1,cluster-2`.
- What is the visibility of the virtual volumes to be added?

Some properties of virtual volumes and consistency groups limit which volumes can be added to a consistency group, or prevent a property of the consistency group from being modified.

For example, a consistency group's `visibility` property is set to `cluster-1`. Virtual volumes local to cluster-1 are added. The `visibility` property of the consistency group cannot be changed to either `cluster-2` or `cluster-1,cluster-2` since the volumes are not visible at cluster-2.

To create a consistency group and to configure those properties that should be set before virtual volumes are added, perform the following:

Steps

1. Use the `ls /clusters/*/consistency-groups/` command to display the names of all the consistency groups:

```
VPlexcli:/> ls /clusters/*/consistency-groups/
/clusters/cluster-1/consistency-groups:
TestCG          local_test      test10      test11      test12  test13  test14
test15          test16          test5       test6       test7   test8   test9
vs_RAM_c1wins   vs_RAM_c2wins   vs_oban005  vs_sun190
/clusters/cluster-2/consistency-groups:
TestCG          local_test      test10      test11      test12  test13  test14
test15          test16          test5       test6       test7   test8   test9
vs_RAM_c1wins   vs_RAM_c2wins   vs_oban005  vs_sun190
```

2. Use the `consistency-group create` command to create a consistency group on a cluster. Specify a name for the new consistency group that did not appear in the output in the previous step.

```
VPlexcli:/> consistency-group create --name TestCG --cluster cluster-1
```

3. Use the `ls /clusters/cluster-id/consistency-groups/`*consistency-group/* command to display the new consistency group.

   Setting the visibility property

   By default, the consistency group's `visibility` property is set to the cluster where the consistency group was created. If a consistency group is created on cluster-2, it is initially visible only on cluster-2.

   Visibility can be configured as follows:

   - `cluster-1` - volumes local to cluster-1.
   - `cluster-2` - volumes local to cluster-2.
   - `cluster-1,cluster-2` - volumes that are distributed with legs at both clusters.

4. Use the `set` command to configure the consistency group's visibility property.

   CAUTION: The CLI context of the consistency group appears only at the cluster where the consistency group has visibility. If visibility is set from cluster-1 to include only cluster-2, the CLI context for the consistency group disappears at cluster-1 and is visible only from cluster-2.

To set the consistency group's visibility property to both clusters:

```
VPlexcli:/clusters/cluster-1/consistency-groups> set TestCG::visibility
cluster-1,cluster-2
```

To set the consistency group's visibility property to cluster-1:

```
VPlexcli:/clusters/cluster-1/consistency-groups> set TestCG::visibility cluster-1
```

To set the consistency group's visibility property to cluster-2:

```
VPlexcli:/clusters/cluster-1/consistency-groups> set TestCG::visibility cluster-2
```

Setting the storage-at-clusters property

By default, the consistency group's `storage-at-clusters` property is set to empty.

The `storage-at-clusters` field tells metro node at which cluster the physical storage that is associated with a consistency group is located. If `visibility` is set to one cluster, then `storage-at-clusters` must be the same as visibility. If `visibility` is set to two clusters (1 and 2), then `storage-at-clusters` can be one of:

- `cluster-1`
- `cluster-2`
- `cluster-1,cluster-2`

  A volume that does not have local storage at every cluster specified by the storage-at-clusters property of a consistency group, cannot be added to the consistency group.

5. Use the set command to configure the consistency group's storage-at-clusters property

```
VPlexcli:/clusters/cluster-1/consistency-groups> set TestCG::storage-at-clusters
cluster-1,cluster-2
```

6. Optionally, use one of the `consistency-group set-detach-rule` commands to apply a detach rule.

   For example, configure the detach-rule as `active-cluster-wins`:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set-detach-rule active-
cluster-wins
```

7. Use the `ll` command to display the new consistency group.

   See Table 16 Consistency group field descriptions for descriptions of the fields in the display.

Adding volumes to a consistency group

A maximum number of 1000 volumes can be added to a consistency group.

About this task

All volumes used by the same application should be grouped together in a consistency group.

Only local volumes can be added to synchronous consistency groups with visibility and storage-at-cluster set to the local cluster.

Remote volumes can be added to synchronous consistency groups with visibility set to both clusters and storage-at-cluster set to one cluster.

Distributed volumes can be added to synchronous consistency groups with visibility set to both clusters and storage-at-cluster set to both clusters.

To add virtual volumes to an existing consistency group, do the following:

Steps

1. Navigate to the target consistency group's context:

```
VPlexcli:/> cd clusters/cluster-1/consistency-groups/TestCG
```

2. Use the `consistency-group list-eligible-virtual-volumes` command to display virtual volumes that are eligible to be added to the consistency group:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> consistency-group list-
eligible-virtual-volumes
[TestDDevice-1_vol, TestDDevice-2_vol, TestDDevice-3_vol, TestDDevice-4_vol,
TestDDevice-5_vol]
```

3. Use the `add-virtual-volumes` command to add virtual volumes to the consistency group.

   To add a single virtual volume:

```
VPlexcli:/clusters/cluster-2/consistency-groups/TestCG> add-virtual-volumes --virtual-
volumes TestDDevice-2_vol
```

   NOTE: The full path is not required if the volume name is unique in the metro node.

   To add multiple volumes using a single command, separate virtual volumes by commas:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> add-virtual-volumes
TestDDevice-1_vol,TestDDevice-2_vol
```

4. Use the `ll` command to display the change.

Removing volumes from a consistency group

To remove one or more virtual volumes from a consistency group:

Steps

1. Use the `ll` command to display the virtual volumes in the target consistency group:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> ll
Attributes:
Name                  Value
--------------------  ------------------------------------------------------------
active-clusters       []
cache-mode            synchronous
detach-rule           winner cluster-1 10s
operational-status    [(cluster-1,{ summary:: ok, details:: [] }), (cluster-2,{
                      summary:: ok, details:: [] })]
passive-clusters      [cluster-1, cluster-2]
recoverpoint-enabled  false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       [TestDDevice-1_vol, TestDDevice-2_vol,
                      TestDDevice-3_vol,TestDDevice-4_vol,
                      TestDDevice-5_vol]
visibility            [cluster-1, cluster-2]
Contexts:
Name          Description
------------  -----------
advanced      -
recoverpoint  -
```

2. Use the `consistency-group remove-virtual-volumes` command to remove one or more virtual volumes.

```
VPlexcli:/> consistency-group remove-virtual-volumes /clusters/cluster-1/virtual-
volumes/TestDDevice-2_vol, --consistency-group /clusters/cluster-1/consistency-groups/
TestCG
```

To remove multiple virtual volumes with a single command, separate the volumes using commas:

```
VPlexcli:/> consistency-group remove-virtual-volumes /clusters/cluster-1/virtual-
volumes/TestDDevice-2_vol, /clusters/cluster-1/virtual-volumes/TestDDevice-3_vol --
consistency-group /clusters/cluster-1/consistency-groups/TestCG
```

Remove two virtual volumes from the target consistency group context:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> remove-virtual-volumes
TestDDevice-2_vol, TestDDevice-3_vol
```

3. Use the `ls` command to display the change:

```
VPlexcli:/> ls clusters/cluster-1/consistency-groups/TestCG
/clusters/cluster-1/consistency-groups/TestCG:
Attributes:
Name                  Value
--------------------  ------------------------------------------------------------
active-clusters       []
cache-mode            synchronous
detach-rule           winner cluster-1 10s
operational-status    [(cluster-1,{ summary:: ok, details:: [] }), (cluster-2,{
                      summary:: ok, details:: [] })]
passive-clusters      [cluster-1, cluster-2]
recoverpoint-enabled false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       [TestDDevice-1_vol, TestDDevice-4_vol, TestDDevice-5_vol]
visibility            [cluster-1, cluster-2]
Contexts:
Name          Description
------------  -----------
advanced      -
recoverpoint  -
```

Modifying consistency group properties

About this task

Use the consistency group set-detach rules to modify the Detach-rule applied to a consistency group:

- `consistency-group set-detach-rule no-automatic-winner`
- `consistency-group set-detach-rule winner`

Use the `set` command to modify the following properties of a consistency group:

- Visibility
- Storage-at-clusters
- Local-read-override

To display which attributes are modifiable (writable) using the `set` command and their valid inputs:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set
attribute            input-description
-------------------
------------------------------------------------------------------------------------------
--------
active-clusters      Read-only.
cache-mode           Read-only.
detach-rule          Read-only.
name                 Takes a unique, non-empty and non-null name. A valid name starts
with a letter or '_'
                     and contains only letters, numbers, '-' and '_'.
operational-status   Read-only.
passive-clusters     Read-only.
read-only            Takes one of '0', '1', 'f', 'false', 'n', 'no', 'off', 'on', 't',
'true', 'y', 'yes' (not case sensitive).
storage-at-clusters  Takes a list with each element being a 'cluster' context or a
context pattern.
```

```
virtual-volumes      Read-only.
visibility           Takes a list with each element being a 'cluster' context or a
context pattern.
```

To display the current setting of a property:

```
VPlexcli:/> set /clusters/cluster-1/consistency-groups/TestCG::cache-mode
```

To display the default values for the target consistency group:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set --default
attribute             default-value
--------------------  -----------------
active-clusters       No default value.
cache-mode            synchronous.
detach-rule           No default value.
name                  No default value.
operational-status    No default value.
passive-clusters      No default value.
read-only             No default value.
storage-at-clusters   No default value.
virtual-volumes       No default value.
visibility            No default value.
```

Example of modify: set visibility

To change the `visibility` property from the target consistency group context:

About this task

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set visibility
cluster-1,cluster-2
```

To change the `visibility` property from the consistency group context:

```
VPlexcli:/clusters/cluster-1/consistency-groups> set TestCG::visibility
cluster-1,cluster-2
```

To change the `visibility` property from the root context:

```
VPlexcli:/> set /clusters/cluster-1/consistency-groups/TestCG::visibility
cluster-1,cluster-2
```

Example of modify: apply a detach rule

The following table lists the applicable detach-rules for consistency groups with various settings for visibility and storage-at-clusters.

About this task

Table 10. Consistency groups detach-rules and visibility, storage-at-volumes

visibility	storage-at-clusters	Applicable detach-rule settings
cluster-1	cluster-1	N/A
cluster-1 and cluster-2	cluster-1 and cluster-2	• no-automatic-winner • winner cluster-1 • winner cluster-2
cluster-1 and cluster-2	cluster-1	• no-automatic-winner • winner cluster-1

To apply a detach rule that will determine the behavior of all volumes in a consistency group:

Steps

1. Use the `ll` command to display the current detach rule (if any) applied to the consistency group:

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG2> ll
Attributes:
Name                  Value
--------------------  ----------------------
active-clusters       []
cache-mode            synchronous
detach-rule           -
.
.
.
```

2. Use one of the `consistency-group set-detach-rule` commands to apply a detach-rule to the consistency group:
   - Use the `consistency-group set-detach-rule no-automatic-winner` command to set the detach-rule as `no-automatic-winner`.

     In the following example, the command is used in the target consistency group's context:

     ```
     VPlexcli:/clusters/cluster-1/consistency-groups/TestCG> set-detach-rule no-
     automatic-winner
     ```

   - Use the `consistency-group set-detach-rule winner` command to specify which cluster is the winner, and the number of seconds metro node waits after a link outage before detaching the winning cluster.

     In the following example, the command is used in the root context:

     ```
     VPlexcli:/> consistency-group set-detach-rule winner --cluster cluster-1 --delay 5s
     --consistency-groups TestCG
     ```

Deleting a consistency group

About this task

To destroy an empty consistency group:

Steps

1. Use the `ls -f` command to verify that there are no virtual volumes in the consistency group (`virtual volumes = [ ]`).

```
VPlexcli:/> ls clusters/cluster-1/consistency-groups/TestCG
Attributes:
Name                  Value
--------------------  ----------------------
active-clusters       []
cache-mode            synchronous
detach-rule           -
operational-status    [ok]
passive-clusters      []
recoverpoint-enabled  false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       []
visibility            [cluster-1, cluster-2]
.
.
.
```

2. Use the `consistency-group destroy` command to delete the consistency group.

To delete a consistency group from the root context:

```
VPlexcli:/> consistency-group destroy clusters/cluster-1/consistency-groups/TestCG
WARNING: The following items will be destroyed:
Context
------------------------------------------
/clusters/cluster-1/consistency-groups/TestCG
Do you wish to proceed? (Yes/No)Yes
```

To delete a consistency group from the consistency group context:

```
VPlexcli:/clusters/cluster-1/consistency-groups> destroy TestCG
WARNING: The following items will be destroyed:
Context
------------------------------------------
/clusters/cluster-1/consistency-groups/TestCG
Do you wish to proceed? (Yes/No)Yes
```

Displaying consistency group properties

You can display the properties of a consistency group.

Use the `ls` in the `/clusters/*/consistency-groups` context to display only the names of consistency groups on all clusters:

```
VPlexcli:/> ls /clusters/*/consistency-groups/
/clusters/cluster-1/consistency-groups:
TestCG         local_test     test10      test11     test12  test13  test14
test15         test16         test5       test6      test7   test8   test9
vs_RAM_c1wins  vs_RAM_c2wins  vs_oban005  vs_sun190
/clusters/cluster-2/consistency-groups:
TestCG         local_test     test10      test11     test12  test13  test14
test15         test16         test5       test6      test7   test8   test9
vs_RAM_c1wins  vs_RAM_c2wins  vs_oban005  vs_sun190
```

Use the `ls` command in the `/clusters/`*`cluster-name`*`/consistency-groups` context to display the names of consistency groups only on the specified cluster:

```
VPlexcli:/> ls /clusters/cluster-1/consistency-groups/
/clusters/cluster-1/consistency-groups:
TestCG      test10     test11  test12  test13  test14  test15  test16  test5  test6
test7  test8  test9  vs_RAM_c1wins  vs_RAM_c2wins
vs_oban005  vs_sun190
```

Use the `ll` command in the `/clusters/`*`cluster-name`*`/consistency-groups` context to display an overview of the consistency groups.

Use this command to monitor the overall health of consistency groups and identify poorly-configured rules:

```
VPlexcli:/clusters/cluster-1/consistency-groups> ll
Name                Operational Status            Active      Passive     Detach Rule
       Cache Mode
------------------  ----------------------------  Clusters    Clusters
------------------  ------------
------------------  ----------------------------  ----------  ----------
------------------  ------------
D850-008_view1      (cluster-1,{ summary:: ok,    cluster-1   cluster-2   active-
cluster-wins  synchronous
                    details:: [] }),
                    (cluster-2,{ summary:: ok,
                    details:: [] })
D850-008_view2      (cluster-1,{ summary:: ok,                cluster-1,  active-
cluster-wins  synchronous
                    details:: [] }),                          cluster-2
                    (cluster-2,{ summary:: ok,
                    details:: [] })
RAM_LR_cluster-1    (cluster-1,{ summary:: ok,                             -
       synchronous
```

```
                     details:: [] }),
                     (cluster-2,{ summary::
                     unknown, details:: [] })
RAM_RR_cluster-2     (cluster-1,{ summary:: ok,                              no-automatic-
winner  synchronous
                     details:: [] }),
                     (cluster-2,{ summary:: ok,
                     details:: [] })
.
.
.
```

Use the `ls` command in the `/clusters/`*`cluster-name`*`/consistency-groups/`*`consistency-group`* context to display the operational status of the groups.

In the following example, the command displays the operational status of a consistency group on a healthy metro node:

```
VPlexcli:/> ls /clusters/cluster-1/consistency-groups/cg1
/clusters/cluster-1/consistency-groups/cg1:
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                  Value
--------------------  ------------------------------------------------------------
active-clusters       [cluster-1, cluster-2]
cache-mode            synchronous
detach-rule           no-automatic-winner
operational-status    [(cluster-1,{ summary:: ok, details:: [] }),
                       (cluster-2,{ summary:: ok, details:: [] })]
passive-clusters      []
read-only             false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       [dd1_vol, dd2_vol]
visibility            [cluster-1, cluster-2]
Contexts:
Name          Description
------------  -----------
advanced      -
```

Use the `ll` command in the `/advanced` context of the consistency group to display the advanced properties of a specified consistency group.

```
VPlexcli:/clusters/cluster-1/consistency-groups/TestCG/advanced> ls
Name                         Value
---------------------------- --------
auto-resume-at-loser         true
current-queue-depth          -
current-rollback-data        -
default-closeout-time        -
delta-size                   -
local-read-override          true
max-possible-rollback-data   -
maximum-queue-depth          -
potential-winner             -
write-pacing                 disabled
```

The following example displays output of the `ls` command in the `/clusters/`*`cluster-name`*`/ consistency-groups/`*`consistency-group`* context during an inter-cluster link outage.

- The `detach-rule` is `no-automatic-winner`, so I/O stops at both clusters. metro node remains in this state until either the inter-cluster link restarts, or you intervene using the `consistency-group choose-winner` command.
- Status `summary` is `suspended`, showing that I/O has stopped.
- Status `details` contain `cluster-departure`, indicating that the clusters can no longer communicate with one another.

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                  Value
--------------------  ------------------------------------------------------------
active-clusters       [cluster-1, cluster-2]
cache-mode            synchronous
detach-rule           no-automatic-winner
operational-status    [(cluster-1,{ summary:: suspended, details:: [cluster-departure]
```

```
}),
                      (cluster-2,{ summary:: suspended, details:: [cluster-departure]
})]
passive-clusters     []
recoverpoint-enabled false
storage-at-clusters  [cluster-1, cluster-2]
virtual-volumes      [dd1_vol, dd2_vol]
visibility           [cluster-1, cluster-2]
Contexts:
advanced  recoverpoint
```

- The `ls` command shows consistency group cg1 as suspended, `requires-resume-at-loser` on cluster-2 after cluster-2 is declared the losing cluster during an inter-cluster link outage.
- The `resume-at-loser` command restarts I/O on cluster-2.
- The `ls` command displays the change in operational status:

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                  Value
--------------------  ------------------------------------------------------------
active-clusters       [cluster-1, cluster-2]
cache-mode            synchronous
detach-rule           no-automatic-winner
operational-status    [(cluster-1,{ summary:: ok, details:: [] }),
                       (cluster-2,{ summary:: suspended, details:: [requires-resume-at-
loser] })]
passive-clusters      []
recoverpoint-enabled false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       [dd1_vol, dd2_vol]
visibility            [cluster-1, cluster-2]
Contexts:
advanced  recoverpoint
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> resume-at-loser -c cluster-2
This may change the view of data presented to applications at cluster cluster-2. You
should first stop applications at that cluster. Continue? (Yes/No) Yes
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                  Value
--------------------  ------------------------------------------------------------
active-clusters       [cluster-1, cluster-2]
cache-mode            synchronous
detach-rule           no-automatic-winner
operational-status    [(cluster-1,{ summary:: ok, details:: [] }),
                       (cluster-2,{ summary:: ok, details:: [] })]
passive-clusters      []
recoverpoint-enabled false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       [dd1_vol, dd2_vol]
visibility            [cluster-1, cluster-2]
Contexts:
advanced  recoverpoint
```

Table 11. Consistency group field descriptions

Property	Description
Standard properties	
`cache mode`	`synchronous` (default) - Writes are done synchronously. Writes are not acknowledged to a host unless they have been sent to back-end storage at all clusters.
`detach-rule`	Policy for automatically picking a winning cluster when there is an inter-cluster link outage. A winning cluster is intended to resume I/O operations when the link fails. • `no-automatic-winner` - The consistency group does not select a winning cluster.

Table 11. Consistency group field descriptions (continued)

Property	Description
	• `winner` - The cluster specified by *cluster-name* will be declared the winner if an inter-cluster link outage lasts more than the number of seconds specified by *delay*.

`storage-at-clusters`	The cluster where the physical storage associated with a consistency group is located.
	• Modifiable using the `set` command. If cluster names are cluster-1 and cluster-2 valid values are:
	  ○ `cluster-1` - Storage associated with this consistency group is located only at cluster-1.
	  ○ `cluster-2` - Storage associated with this consistency group is located only at cluster-2.
	  ○ `cluster-1,cluster-2` - Storage associated with this consistency group is located at both cluster-1 and cluster-2.
	• When modified, the new value cannot be incompatible with the volumes that are already in the consistency group. Change `storage-at-clusters` only when the consistency group has no member volumes.

`visibility`	Lists the clusters at which this consistency group is visible.
	• Modifiable using the `set` command. If cluster names are cluster-1 and cluster-2 valid values are:
	  ○ `cluster-1` - This consistency group is visible only at cluster-1.
	  ○ `cluster-2` - This consistency group is visible only at cluster-2.
	  ○ `cluster-1,cluster-2` - This consistency group is visible at both cluster-1 and cluster-2.
	• Changing this property changes where the consistency group is visible, and may cause contexts to appear or disappear in the context tree.

`virtual-volume`	Lists the virtual volumes that are members of the consistency group. Modifiable using the following commands: • `consistency-group add-virtual-volumes` - Add one or more virtual volumes to a consistency group. • `consistency-group remove-virtual-volumes` - Remove one or more virtual volumes from a consistency group.
Advanced properties	
`auto-resume-at-loser`	Determines whether I/O automatically resumes at the detached cluster for the volumes in a consistency group when the cluster regains connectivity with its peer cluster. • Relevant only for multi-cluster consistency groups that contain distributed volumes. • Modifiable using the `set` command. Set this property to `true` to allow the volumes to resume I/O without user intervention (using the `resume-at-loser` command). • `true` - I/O automatically resumes on the losing cluster after the inter-cluster link has been restored. • `false` (default) - I/O must be resumed manually after the inter-cluster link has been restored. • Leave this property set to `false` to give administrators time to restart the application. Otherwise, dirty data in the host's cache is not consistent with the image on disk to which the winning cluster has been actively writing. Setting this property to `true` can cause a spontaneous change of the view of data presented to applications at the losing cluster. Most applications cannot tolerate this data change. If the host flushes those dirty pages out of sequence, the data image may be corrupted.
Display-only properties	
`active-clusters`	For synchronous consistency groups, this property is always empty ([ ]).

Table 11. Consistency group field descriptions (continued)

Property	Description
`operational status`	Current status for this consistency group with respect to each cluster on which it is visible. ● `ok` - I/O can be serviced on the volumes in the consistency group. ● `suspended` - I/O is suspended for the volumes in the consistency group. The reasons are described in `operational status: details`. ● `degraded` - I/O is continuing, but there are other problems as described in `operational status: details`. ● `unknown` - The status is unknown, mostly because of lost management connectivity.
`operational status: details`	If `operational status` is `ok`, this field is empty: `[ ]`. Otherwise, it displays additional information, which may be any of the following: ● `cluster-departure` - Not all the visible clusters are in communication. ● `data-safe-failure` - A single director has failed. The volumes are still crash-consistent, and remain in this state, unless a second failure occurs before the first is recovered. ● `rebuilding-across-clusters` - One or more distributed member volumes is being rebuilt. At least one volume in the group is out of date at that cluster and is re-syncing. If the link goes out at this time, the entire consistency group is suspended. Use the `rebuild status` command to display which volume is out of date at which cluster. ● rebuilding-within-cluster - One or more local rebuilds is in progress at this cluster. ● `requires-resolve-conflicting-detach` - After the inter-cluster link is restored, two clusters have discovered that they have detached from one another and resumed I/O independently. The clusters are continuing to service I/O on their independent versions of the data. The `consistency-group resolve-conflicting-detach` command must be used to make the view of data consistent again at the clusters. ● `requires-resume-after-rollback` - A cluster has detached its peer cluster and rolled back the view of data, but is awaiting the `consistency-group resume-after-rollback` command before resuming I/O. Displayed: ○ There is no detach-rule ○ If the detach-rule is `no-automatic-winner`, or ○ If the detach-rule cannot fire because its conditions are not met. ■ `unhealthy-devices` - I/O has stopped in this consistency group because one or more volumes are unhealthy and cannot perform I/O. ■ `will-rollback-on-link-down` - If there were a link-down now, the winning cluster would have to roll back the view of data in order to resume I/O.
`virtual-volumes`	List of virtual volumes that are members of the consistency group.

Operating a consistency group

Best practice is to allow I/O to continue at only one cluster. Allowing I/O to continue at both clusters will result in the complete resync of one cluster from the other. All writes at the loser cluster are lost.

About this task

When I/O continues at both clusters:

- The data images at the clusters diverge.
- Legs of distributed volumes are logically separate.

When the inter-cluster link is restored, the clusters learn that I/O has proceeded independently. I/O continues at both clusters until you pick a winning cluster whose data image will be used as the source to synchronize the data images.

In the following example, I/O resumed at both clusters during an inter-cluster link outage. When the inter-cluster link is restored, the two clusters come back into contact and learn that they have each detached the other and carried on I/O.

Steps

1. Use the `ls` command to display the consistency group's operational status at both clusters.

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                  Value
--------------------  ----------------------------------------
active-clusters       [cluster-1, cluster-2]
cache-mode            synchronous
detach-rule           no-automatic-winner
operational-status    [(cluster-1,{ summary:: ok, details:: [requires-resolve-
conflicting-detach] }),
                      (cluster-2,{ summary:: ok, details:: [requires-resolve-conflicting-detach
passive-clusters      []
recoverpoint-enabled  false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       [dd1_vol, dd2_vol]
visibility            [cluster-1, cluster-2]
Contexts:
advanced  recoverpoint
```

2. Use the `resolve-conflicting-detach` command to select cluster-1 as the winner.

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> resolve-conflicting-detach -c
cluster-1
This will cause I/O to suspend at clusters in conflict with cluster cluster-1,
allowing you to stop applications at those clusters. Continue? (Yes/No) Yes
```

   Cluster-2's modifications to data on volumes in the consistency group since the link outage started are discarded.

   Cluster-2's data image is then synchronized with the image at cluster-1.

   I/O gets suspended at cluster-2 if the auto-resume policy is false.

3. Use the `ls` command to verify the change in operation status:

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                  Value
--------------------  ------------------------------------------------------------
active-clusters       [cluster-1, cluster-2]
cache-mode            synchronous
detach-rule           no-automatic-winner
operational-status    [(cluster-1,{ summary:: ok, details:: [] }),
                       (cluster-2,{ summary:: suspended, details:: [requires-resume-at-
loser] })]
passive-clusters      []
recoverpoint-enabled  false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       [dd1_vol, dd2_vol]
visibility            [cluster-1, cluster-2]
Contexts:
advanced  recoverpoint
```

   - At cluster-1, I/O continues, and the status is `ok`.
   - At cluster-2, the view of data has changed and hence I/O is suspended.

4. Use the `consistency-group resume-at-loser` command to resume I/O to the consistency group on cluster-2.

Resuming I/O after rollback

About this task

Without that data, the winning cluster's data image is inconsistent. Resuming I/O at the winner requires rolling back the winner's data image to the last point where the clusters agreed.

This can cause a sudden change in the data image.

Many applications cannot tolerate sudden data changes, so the roll-back and resumption of I/O requires manual intervention.

The delay gives the administrator the chance to halt applications before changing the data image. The data image is rolled back as soon as a winner is chosen (either manually or automatically using a detach rule).

The resume-after-rollback command acknowledges that the application is ready for recovery (this may involve application failure and/or restarting the host).

NOTE: It is recommended to restart the hosts of affected applications.

Steps

1. Use the `ls` command to display the consistency group on the winning cluster during an inter-cluster link outage.

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                   Value
---------------------  ------------------------------------------
active-clusters        []
cache-mode             synchronous
detach-rule            -
operational-status     [suspended, requires-resume-after-rollback]
passive-clusters       [cluster-1, cluster-2]
recoverpoint-enabled   false
storage-at-clusters    [cluster-1, cluster-2]
virtual-volumes        [dd1_vol]
visibility             [cluster-1, cluster-2
Contexts:
advanced  recoverpoint
```

2. Use the `resume-after-rollback` command to acknowledge that the application is ready for recovery.

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> resume-after-rollback --
consistency-group cg1
This will change the view of data at cluster cluster-1, so you should ensure
applications are stopped at that cluster. Continue? (Yes/No) Yes
```

3. Use the `ls` command to display the change in operational status.

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                   Value
---------------------  ----------------------
active-clusters        [cluster-1]
cache-mode             synchronous
detach-rule            -
operational-status     [ok]
passive-clusters       [cluster-2]
recoverpoint-enabled   false
storage-at-clusters    [cluster-1, cluster-2]
virtual-volumes        [dd1_vol]
visibility             [cluster-1, cluster-2]
Contexts:
advanced  recoverpoint
```

Resuming I/O at the losing cluster

During an inter-cluster link outage, an you might permit I/O to resume at one of the two clusters, the *winning* cluster.

About this task

I/O remains suspended on the *losing* cluster.

When the inter-cluster link restores, the winning and losing clusters re-connect, and the losing cluster discovers that the winning cluster has resumed I/O without it.

Unless explicitly configured, I/O remains suspended on the losing cluster. This prevents applications at the losing cluster from experiencing a spontaneous data change.

The delay allows you to shut down applications.

After stopping the applications, use the `consistency-group resume-at-loser` command to:

- Resynchronize the data image on the losing cluster with the data image on the winning cluster.
- Resume servicing I/O operations.

You can then safely restart the applications at the losing cluster.

To restart I/O on the losing cluster:

Steps

1. Use the `ls` command to display the operational status of the target consistency group.

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                  Value
--------------------  ------------------------------------------------------------
active-clusters       [cluster-1, cluster-2]
cache-mode            synchronous
detach-rule           no-automatic-winner
operational-status    [(cluster-1,{ summary:: ok, details:: [] }),
                       (cluster-2,{ summary:: suspended, details:: [requires-resume-at-
loser] })]
passive-clusters      []
recoverpoint-enabled false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       [dd1_vol, dd2_vol]
visibility            [cluster-1, cluster-2]
Contexts:
advanced  recoverpoint
```

2. Use the `consistency-group resume-at-loser` to restart I/O on the losing cluster.

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> resume-at-loser -c cluster-2
This may change the view of data presented to applications at cluster cluster-2. You
should first stop applications at that cluster. Continue? (Yes/No) Yes
```

3. Use the `ls` command to verify the change in operational status:

```
VPlexcli:/clusters/cluster-1/consistency-groups/cg1> ls
Attributes:
Name                  Value
--------------------  ------------------------------------------------------------
active-clusters       [cluster-1, cluster-2]
cache-mode            synchronous
detach-rule           no-automatic-winner
operational-status    [(cluster-1,{ summary:: ok, details:: [] }),
                       (cluster-2,{ summary:: ok, details:: [] })]
passive-clusters      []
recoverpoint-enabled false
storage-at-clusters   [cluster-1, cluster-2]
virtual-volumes       [dd1_vol, dd2_vol]
visibility            [cluster-1, cluster-2]
```

```
Contexts:
advanced  recoverpoint
```

You might notice `rebuilding-across-clusters` in operational status while devices are rebuilding.

Setting the read-only attribute

SRDF R2 devices (replicas) are an example of an array-managed Business Continuance Volume (BCV). For consistency groups that contain these volumes, you can use the `set` command to set the consistency group to read-only.

About this task

If the read-only attribute is true, the system prevents write operations to virtual volumes in the consistency group. Virtual volumes in a read-only consistency group must be local, and you must map each virtual volume one-to-one to a single storage volume (for example, local RAID 0 with no slicing).

You cannot add virtual volumes with an invalid topology to a read-only consistency group. The `consistency-group add-virtual-volumes` command fails. If you set a consistency group to read-only and that consistency group already contains virtual volumes with an invalid topology, the `set read-only true` command fails.

A consistency group cannot be `read-only` and `recoverpoint-enabled` at the same time, since the two properties are incompatible.

Steps

Use the `set` command to set the consistency group to read-only.

```
VPlexcli:/> cd/clusters/cluster-1/consistency-groups/test
VPlexcli:/clusters/cluster-1/consistency-groups/test>set read-only true
VPlexcli:/clusters/cluster-1/consistency-groups>ll
Name   Operational Active  Passive  Detach Rule  Cache Mode Read
------ Status     Clusters Clusters ----------  ---------   Only
------- ----------- ------- -------- ---------- --------- ----
DB2_app (Hopkinton,{ winner Hopkinton after 5s synchronous true
         summary:: ok,
         details:: []
         }),
         Providence, {
         summary:: ok,
         details:: []
         })
```

10

Performance and Monitoring

This chapter describes RPO/RTO and the procedures to create and operate performance monitors.

Topics:

About performance

This chapter describes the following topics related to performance on metro node systems:

- Configuration - Modifiable parameters to maximize performance and to manage Recovery Point Objective (RPO) and Recovery Time Objective (RTO).
- Monitoring - Tools and techniques to monitor the performance of metro node, and to identify and diagnose problems.

RPO and RTO

Recovery Point Objective (RPO): RPO is the time interval between the point of failure of a storage system and the expected point in the past to which the storage system is capable of recovering customer data.

RPO is a maximum amount of data loss that can be tolerated by the application after a failure. The value of the RPO is highly dependent upon the recovery technique used. For example, RPO for backups is typically days; for asynchronous replication minutes; and for mirroring or synchronous replication seconds or instantaneous.

Recovery Time Objective (RTO): RTO is the time duration within which a storage solution is expected to recover from failure and begin servicing application requests.

RTO is the longest tolerable application outage due to a failure of a storage system. RTO is a function of the storage technology. It may measure in hours for backup systems, minutes for a remote replication, and seconds (or less) for a mirroring.

About performance monitoring

Performance monitors collect and displays statistics to determine how a port or volume is being used, how much I/O is being processed, CPU usage, and so on.

Performance monitoring is supported in both the metro node CLI and Unisphere, and falls into three general types:

- Current load monitoring allows administrators to watch CPU load during upgrades, I/O load across the inter-cluster WAN link, and front-end compared to back-end load during data mining or back up.

  Current load monitoring is supported in Unisphere.

- Long term load monitoring collects data for capacity planning and load balancing.

  Long term load monitoring is supported by monitors created in the CLI and/or perpetual monitors.

- Troubleshooting monitoring helps identify bottlenecks and resource hogs.

  Troubleshooting monitors are supported by monitors created in the CLI and/or perpetual monitors.

NOTE: In Unisphere for metro node, performance statistics are displayed per cluster. To view statistics for both clusters in a Metro configuration, connect to both clusters.

Custom monitors

You can use the CLI to create custom monitors to collect and display selected statistics for selected targets.

See Monitor performance using the CLI.

Perpetual monitors

GeoSynchrony includes perpetual monitors that gather a standard set of performance statistics every 30 seconds. Perpetual monitors collect the statistics related to the performance of metro node directors and virtual volumes.

Perpetual monitor files are collected as part of `collect-diagnostics`. Collect-diagnostics is per cluster, so in Metro configurations, run the command from both metro node management servers.

Output of perpetual monitors is captured in the file `smsDump_date.zip` inside the base collect-diagnostics zip file.

Within `smsDump_date.zip` file, monitor files are in `clilogs/`.

You can also copy the perpetual files from the management server. They are located in `/var/log/VPlex/cli/`. There is one perpetual monitor file per director, identifiable by the keyword “PERPETUAL” .

The following is an example for the statistics that perpetual monitors collect on virtual volumes:

```
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.1
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.2
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.3
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.4
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.5
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.6
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.7
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.8
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.9
director-1-1-A_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.10
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.1
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.2
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.3
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.4
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.5
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.6
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.7
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.8
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.9
director-1-1-B_VIRTUAL_VOLUMES_PERPETUAL_MONITOR.log.10
```

Performance monitoring using Unisphere for metro node

The performance monitoring dashboard provides a customized view into the performance of your system. You decide which aspects of the system's performance to view and compare.

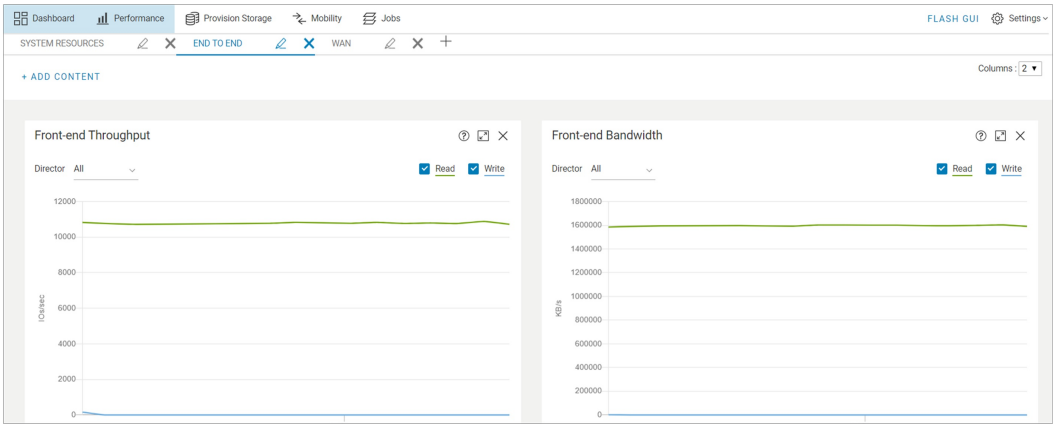

Figure 9. Performance monitoring dashboard (for HTML5)

Performance information for the current 5-minute window is displayed as a set of charts, including:

- **WAN Link Performance chart** - Shows the WAN link performance for the cluster to you are connected to. Use this chart to monitor link performance to help determine the bandwidth requirements for your specific environment, gather statistical data over time, monitor network traffic during peak periods, or to plan data mobility jobs to avoid peak usage times.
- **WAN Latency chart** - Provides a time-based view of the WAN Latency. The categories avg-lat/min-lat/max-lat each report values observed in the last 5 seconds or less.
- **Write Latency Delta chart** - Provides the delta between Front-end latency and Back-end Latency per director. This is a key metric for Local/Metro – the amount of overhead time metro node spends processing a write.
- **Back-end Errors chart** - Displays the back-end I/O errors to and from the storage array. There are three categories of back-end errors: Aborts, timeouts, and resets.
- **Back-end Throughput chart** - Shows the back-end I/Os per second over time for directors. Generally throughput (or more commonly referred to as IOPS) is associated with small block I/O (4KB or 16KB I/O requests.)
- **Back-End Bandwidth chart** - Shows the quantity of back-end reads and writes per second over time for directors. Generally bandwidth (measured in KB/s or MB/s) is associated with large block I/O (64KB or greater I/O requests).
- **Back-end Latency chart** - Provides details of the back-end latency statistics for your metro node system in graphical form over time. The chart allows you to view current or historical performance data that you can use to monitor peaks in workload, detect performance issues, or view what was happening in the system when a specific problem occurred.
- **Rebuild Status dashboard** - Display the status of any rebuilds or migration operations that are running on your metro node system.
- **CPU Utilization chart** - Provides a time-based view of the utilization load on the primary director CPU on your metro node system. By default, the chart shows an averaged view of the utilization loads of all the directors in your metro node system.
- **Heap Usage chart** - Shows a percentage of the heap memory used by the firmware on a director.
- **Front-end Aborts chart** - Displays the number of aborts per second over time for directors on your metro node system. By default, the chart shows averaged front-end aborts for the metro node system.
- **Front-End Bandwidth chart** - Displays the quantity of front-end reads and writes per second over time for directors on your metro node system. By default, the chart shows the total front-end bandwidth for the metro node system.
- **Front-end Latency chart** - Provides details of the front-end latency statistics for your metro node system in graphical form over time. The chart allows you to view current or historical performance data that you can use to monitor peaks in workload, detect performance issues, or view what was happening in the system when a specific problem occurred.
- **Front-end Queue Depth chart** - Provides the count of front-end operations per director. It describes the number of concurrent outstanding operations active in the system.
- **Front-End Throughput chart** - Displays the front-end I/Os per second over time for directors on your metro node system. By default, the chart shows the total front-end throughput for the metro node system.
- **Virtual Volume Throughput chart** - Provides a time-based view of the total throughput or IOPS for a virtual volume. Generally throughput, more commonly referred to IOPS, is associated with small block I/O (512B to 16KB I/O requests.
- **Virtual Volume Latency chart** - Provides a time-based view of the IO Latency for a virtual volume broken down by read and write latency. Virtual volume latency is defined as the amount of time an I/O spends within metro node for a given virtual volume.

- **Virtual Volume Bandwidth chart** - Provides a time-based view of the total bandwidth (or KB/s or MB/s) in reads and writes for a virtual-volume. Generally bandwidth (also referred to as KB/s or MB/s), is associated with large block I/O (64KB or greater I/O requests)
- **Front-end ports dashboard** - Displays performance metrics for all metro node front-end ports. The dashboard does not provide historical data, but refreshes every five seconds and displays data from the previous five-second period.

Performance monitoring using the metro node CLI

Use the CLI to create custom monitors to help diagnose performance issues.

Two CLI objects collect and display performance statistics:

- *monitors* - Gather the specified statistic from the specified target at the specified interval.
- *monitor sinks* - Direct the output to the desired destination. Monitor sinks include the console, a file, or a combination of the two.

Monitor performance using the CLI

This section describes the steps to create a custom monitor using the metro node CLI.

About file rotation and timestamps

The log files created by a monitor's file sink are automatically rotated when they reach a size of 10 MB. The 10MB file is saved as `filename.csv.n` where *n* is a number 1 - 10, and output is saved in a new file named `filename.csv.n+1`.

The `.csv` files are rotated up to 10 times.

In the following example, a monitor has exceeded 10MB of output. The initial 10MB are stored in *`filename`*`.csv.1`. Subsequent output is stored in *`filename`*`.csv`.

```
service@sms-cluster-1:/var/log/VPlex/cli> ll my-data.csv*
-rw-r--r-- 1 service users 2910722 2012-03-06 21:23 my-data.csv
-rw-r--r-- 1 service users 10566670 2012-03-06 21:10 my-data.csv.1
```

If the second file exceeds, 10MB:

- The previous *`filename`*`.csv.1` is changed to *`filename`*`.csv.2`
- The *`filename`*`.csv` changes to *`filename`*`.csv.1`
- Subsequent output is stored in *`filename`*`.csv`

Up to 10 such rotations, and numbered `.csv` files are supported.

When the file sink is removed or the monitor is destroyed, output to the `.csv` file stops, and the current `.csv` file is is timestamped. For example:

```
service@sms-cluster-1:/var/log/VPlex/cli> ll my-data.csv*
-rw-r--r-- 1 service users 10566670 2012-03-06 21:23 my-data.csv.1
-rw-r--r-- 1 service users  5637498 2012-03-06 21:26 my-data.csv_20120306092614973
```

Procedure overview: create a monitor using the CLI

To create and operate a monitor using the CLI, use the following general steps:

1. Determine the type of statistic to collect from the target object.

   Use the `monitor stat-list category` or the `monitor stat-list *` command to display the statistics to include in the monitor.

   Refer to the tables in Statistics for lists of statistics by category.

   Note whether the statistic you want to collect requires a target to be specified.

Specify only one type of target per monitor. For example, you cannot create a monitor that includes both port and storage volumes as targets.

2. Determine how often the monitor should collect statistics.
3. Use the `monitor create` command to create a monitor.
4. Use the `monitor add-sink` commands to add one or more sinks to the monitor.
   - Add a console sink to send performance data to the metro node management console.
   - Add a file sink to send performance data to a specified file.
5. Repeat Steps 3 and 4 for each director.
6. The monitor begins operation (polling and collecting performance data) when the sink is added to the monitor.

   To disable automatic polling without deleting the monitor or its sinks, do one of the following:

   - Use the `set` command to change the monitor's period attribute to 0.
   - Use the `set` command to change the sink's enabled attribute to false.
7. Use the `monitor collect` command to update and collect statistics immediately without waiting for the monitor's next automatic collection.
8. Monitor output.

   Console sinks display monitor output on the console.

   For file sinks, navigate to `/var/log/VPlex/cli/` on the management server and use the `tail -f` *filename* to display the output,

   or:

   Send output to a csv file, open the file in Microsoft Excel and create a chart.

   Do NOT edit the CSV file in Microsoft Excel, and then save the file. Excel removes the seconds field, resulting in duplicate timestamps. Use Excel to look at the CSV files, but don't save any edits.
9. Use the `monitor destroy` command to remove the monitor.

Creating a monitor

Use the `monitor create` command to create a monitor and specify the statistics collected by the monitor.

About this task

See the Online Help for a complete list of available performance monitor statistics.

Create a simple monitor with the default period, and no targets:

```
VPlexcli:/monitoring> monitor create --name TestMonitor --director director-2-1-B --
stats director.fe-read,director.fe-write
Successfully created 1 monitor(s) out of 1.
```

Create a monitor to collect statistics from the director category on /engines/engine-1-1/directors/director-1-1-A every 10 seconds:

```
VPlexcli:/monitoring> monitor create --name DirStats --period 10s --director /clusters/
cluster-1/directors/director-1-1-A --stats director.*
```

Create a monitor to collect statistics on all storage volumes at cluster-1:

```
VPlexcli:/monitoring> monitor create --name SVStats-Cluster1 --director /clusters/
cluster-1/directors/director-1-1-A --stats storage-volume.* --targets /clusters/
cluster-1/storage-elements/storage-volumes/*
```

Create a monitor to collect all front-end statistics on front-end port I0-01:

```
VPlexcli:/monitoring> monitor create --name FE-FC01-stats --director /clusters/cluster-1/
directors/director-1-1-A --stats fe-prt.* --targets /clusters/cluster-1/directors/
director-1-1-A/ports/IO-01
```

Create a perfomance monitor to monitor local COM latency for a specified director:

```
VPlexcli:/> monitor create --name local-cluster --stats "com-cluster-io.*" --director
director-1-1-A --targets "/clusters/cluster-1"
```

Create a perfomance monitor to monitor latency to the remote cluster:

```
VPlexcli:/> monitor create --name remote-cluster --stats "com-cluster-io.*" --director
director-1-1-A --targets "/clusters/cluster-2"
```

Adding/deleting monitor sinks

Every monitor must have at least one sink, and may have multiple sinks. There are two types of sink:

About this task

console - Sends output to metro node management server console.

file - Sends output to the specified file.

Adding a console sink

Use the `monitor add-console-sink` command to add a console sink to an existing monitor.

About this task

Console monitors display the selected statistics on the metro node Management Console, interrupting any other input/output to/from the console. Refer to Enabling/disabling sinks for the command to disable a console sink.

The default format for console sinks is 'table'.

To add a console sink with output formatted as table (the default output format):

VPlexcli:/> monitor add-console-sink --monitor Director-2-1-B_TestMonitor Navigate to the monitor context and use the ll console command to display the sink:

```
VPlexcli:/> cd monitoring/directors/director-2-1-B/monitors/director-2-1-B_TestMonitor/
sinks
VPlexcli:/monitoring/directors/Director-2-1-B/monitors/Director-2-1-B_TestMonitor/sinks>
ll
Name     Enabled  Format  Sink-To
-------  -------  ------  -------
console  true     table   console
VPlexcli:/monitoring/directors/Director-2-1-B/monitors/Director-2-1-B_TestMonitor/sinks>
ll console
/monitoring/directors/Director-2-1-B/monitors/Director-2-1-B_TestMonitor/sinks/console:
Name     Value
-------  -------
enabled  true
format   table
sink-to  console
type     console
```

Adding a file sink

Use the `monitor add-file-sink` command to add a file sink to an existing monitor.

About this task

The default format for file sinks is csv (comma-separated values).

The default name of the new sink is `file`.

The default location for the sink output is `/var/log/VPlex/cli`.

To add a file sink to send output to the specified `.csv` file:

```
VPlexcli:/monitoring/directors/director-1-1-A/monitors> monitor add-file-sink  --monitor
director-1-1-A_stats --file /var/log/VPlex/cli/director_1_1_A.csv
```

Navigate to the monitor sinks context and use the `ll` *sink-name* command to display the sink:

```
VPlexcli:/> cd monitoring/directors/director-1-1-A/monitors/director-1-1-A_stats/sinks
VPlexcli:/monitoring/directors/Director-1-1-A/monitors/director-1-1-A_stats/sinks> ll
file
/monitoring/directors/Director-1-1-A/monitors/director-1-1-A_stats/sinks/file:
Name     Value
-------  ------------------------------
enabled  true
format   csv
sink-to  /var/log/VPlex/cli/director_1_1_A.csv
type     file
```

Deleting a monitor sink

Use the `monitor remove-sink` command to remove a sink from a monitor:

About this task

```
VPlexcli:/monitoring/directors/director-2-1-B/monitors/director-2-1-B_TestMonitor>
monitor remove-sink console
```

Deleting a monitor

Use the `monitor destroy monitor` command to delete a specified monitor.

About this task

For example:

```
VPlexcli:/monitoring/directors/director-1-1-B/monitors> monitor destroy director-1-1-
B_TestMonitor
WARNING: The following items will be destroyed:
Context
------------------------------------------------------------------------------
/monitoring/directors/director-1-1-B/monitors/director-1-1-B_TestMonitor
Do you wish to proceed? (Yes/No) y
```

Create an SNMP monitor

SNMP sinks can be added only to monitors configured to collect fe-lu or disk statistics.

All the statistics in the fe-lu statistics category must be included in the monitor.

In the following example:

- The `monitor stat-list fe-lu` command displays all the statistics in the fe-lu category
- The `monitor create` command creates a monitor to collect all fe-lu statistics
- The `cd` command changes the context to the new monitor
- The `add-snmp-sink` command adds an SNMP sink to the monitor

```
VPlexcli:/monitoring/directors/director-1-1-B/monitors> monitor stat-list fe-lu
Name             Target          Type     Units
---------------  --------------  -------  --------
fe-lu.ops        virtual-volume  counter  counts/s
fe-lu.read       virtual-volume  counter  KB/s
fe-lu.read-lat   virtual-volume  bucket   us
fe-lu.write      virtual-volume  counter  KB/s
fe-lu.write-lat  virtual-volume  bucket   us
VPlexcli:/monitoring/directors/director-1-1-B/monitors> monitor create --name
```

```
SNMPTestMonitor --director director-1-1-B --stats fe-lu.read,fe-lu.read-lat,fe-
lu.write,fe-lu.write-lat,fe-lu.ops  --targets /clusters/cluster-1/virtual-volumes/
polyvol_e4_extent_Symm0487_393
Successfully created 1 monitor(s) out of 1.
VPlexcli:/monitoring/directors/director-1-1-B/monitors> cd director-1-1-
B_SNMPTestMonitor
VPlexcli:/monitoring/directors/director-1-1-B/monitors/director-1-1-B_
SNMPTestMonitor> add-snmp-sink --name fe-lu-stats
Displaying monitors
```

Use the `ls /monitoring/directors/*/monitors` command to display the names of all monitors configured on the system:

```
VPlexcli:/> ls /monitoring/directors/*/monitors
/monitoring/directors/director-1-1-A/monitors:
DEFAULT_director-1-1-A_PERPETUAL_vplex_sys_perf_mon_v8
director-1-1-A_Billy35_FE_A0-FC00_stats
director-1-1-A_director-fe-21112011
director-1-1-A_diskReportMonitor
.
.
.
/monitoring/directors/director-1-1-B/monitors:
DEFAULT_director-1-1-B_PERPETUAL_vplex_sys_perf_mon_v8
.
.
.
```

Use the `ll /monitoring/directors/*/monitors` command to display summarized information about all the monitors for the specified context and object:

```
VPlexcli:/> ll /monitoring/directors/director-1-1-A/monitors
/monitoring/directors/director-1-1-A/monitors:
Name                              Ownership  Collecting  Period  Average  Idle  Bucket
Bucket  Bucket  Bucket
------------------------------    ---------  Data        ------  Period   For   Min
Max     Width   Count
--------------------------------- ---------  ----------  ------  -------  ----  ------
------  ------  ------
director-1-1-A_FE_A0-FC00         false      false       5s      -        -     -       -
     -       64
director-1-1-A_director-fe        false      false       5s      -        -     -       -
     -       64
director-1-1-A_ipcom-21112011     false      false       5s      -        -     -       -
     -       64
director-1-1-A_portReportMon      false      false       5s      -        -     -       -
     -       64
.
.
.
```

Use the `ll /monitoring/directors/*/monitors/`*monitor-name* command to display detailed information about all the specified monitor:

```
VPlexcli: ll /monitoring/directors/director-2-1-B/monitors/director-2-1-
B_volumeReportMonitor
Attributes:
Name              Value
----------------  ------------------------------------------------------------------
average-period    -
bucket-count      64
bucket-max        -
bucket-min        -
bucket-width      -
collecting-data   true
firmware-id       9
idle-for          5.44days
ownership         true
period            0s
statistics        [virtual-volume.ops, virtual-volume.read,
                  virtual-volume.write]
```

```
targets           DR1_C1-C2_1gb_dev10_vol, DR1_C1-C2_1gb_dev11_vol,
                  DR1_C1-C2_1gb_dev12_vol, DR1_C1-C2_1gb_dev13_vol,
                  DR1_C1-C2_1gb_dev14_vol, DR1_C1-C2_1gb_dev15_vol,
                  DR1_C1-C2_1gb_dev16_vol, DR1_C1-C2_1gb_dev17_vol,
                  DR1_C1-C2_1gb_dev18_vol, DR1_C1-C2_1gb_dev19_vol, ... (1300
                  total)
Contexts:
Name   Description
-----  ------------------------------------------------------------------------
sinks  Contains all of the sinks set up to collect data from this performance
       monitor.
```

Use the `ll /monitoring/directors/*/monitors/`*monitor-name*`/sinks` command to display the sinks associated with the specified monitor:

```
VPlexcli: ll /monitoring/directors/director-2-1-B/monitors/director-2-1-
B_volumeReportMonitor/sinks
/monitoring/directors/bob70/monitors/bob70_volumeReportMonitor/sinks:
Name  Enabled  Format  Sink-To
----  -------  ------  ------------------------------------------------------
file  true     csv     /var/log/VPlex/cli/reports/volumeReportMonitor_bob70.csv
```

Table 12. Monitor and sink field descriptions

Field	Description
`average-period`	The actual average sampling period.
`collecting-data`	Whether or not this performance monitor is collecting data. A monitor collects data if it has at least one enabled sink.
`firmware-id`	The firmware ID of the monitor.
`idle-for`	The elapsed time since this performance monitor was accessed in the firmware.
`name`	A director-wide unique name for this performance monitor intended to be meaningful to the user.
`ownership`	Whether or not this monitor was created in this instance of metro node Management Console.
`period`	Sampling period in seconds.
`statistics`	List of performance statistics that are being monitored.
`targets`	List of targets that apply to the monitored performance statistics. A target can be a port, storage volume, or virtual volume. Not all statistics require targets.
Monitor sink display fields	
`Name`	For file sinks, the name of the created sink context. Default is 'file'.
`Enabled`	Whether the monitor sink is enabled or disabled.
`Format`	The required output format. Can be `csv` or `table`. Default is `csv` for file sinks and `table` for console sinks.
`Sink-To`	For file sinks, the filename to sink data to.

Enabling/disabling/changing polling

Polling (collection of the specified statistics) begins when the first sink is added to a monitor. Polling occurs automatically at the interval specified by the monitor's `period` attribute.

About this task

Use the `set` command to change the polling period.

Use the `monitor collect` command to run a collection immediately, before its defined polling interval.

Use the `set` command to disable, or modify automatic polling for a monitor.

In the following example:

- The `set` command changes the period attribute to 0, disabling automatic polling
- The `ll` command displays the change:

```
VPlexcli:/monitoring/directors/director-2-1-B/monitors/director-2-1-B_TestMonitor>
set period 0
VPlexcli:/monitoring/directors/director-2-1-B/monitors/director-2-1-B_TestMonitor> ll
Attributes:
Name              Value
---------------   ------------------------------------------------------------
average-period    -
bucket-count      64
bucket-max        -
bucket-min        -
bucket-width      -
collecting-data   false
firmware-id       4
idle-for          5.78min
ownership         true
period            0s
.
.
.
```

To re-enable polling, use the `set` command to change the period attribute to a non-zero value.

Enabling/disabling sinks

Use the `set` command to enable or disable a monitor sink.

About this task

To disable a monitor sink:

```
VPlexcli:/monitoring/directors/director-2-1-B/monitors/director-2-1-B_TestMonitor/sinks/
console> set enabled false
VPlexcli:/monitoring/directors/director-2-1-B/monitors/director-2-1-B_TestMonitor/sinks/
console> ll
Name     Value
-------  -------
enabled  false
format   table
sink-to  console
type     console
```

To enable a monitor sink:

```
VPlexcli:/monitoring/directors/director-2-1-B/monitors/director-2-1-B_TestMonitor/sinks/
console> set enabled true
VPlexcli:/monitoring/directors/director-2-1-B/monitors/director-2-1-B_TestMonitor/sinks/
console> ll
Name     Value
-------  -------
enabled  true
format   table
sink-to  console
type     console
```

Force an immediate poll

Use the `monitor collect` command to force an immediate poll and collection of performance data without waiting for the automatic poll interval.

For example:

```
VPlexcli:/> monitor collect /monitoring/directors/director-2-1-B/monitors/director-2-1-
B_TestMonitor

Source:                            director-2-1-B_TestMonitor
Time:                                     2010-07-01 10:05:55
director.be-ops (counts/s):
.
.
.
```

Port Monitoring

Details related to `port-stats-monitor` script.

Getting Started

The `port-stats-monitor` script can be used to identify the metro node ports which are observing problems. The metro node system may, or may not, be experiencing an issue due to the problems that the port-stats-monitor script is noting. However, it does indicate that there is a problem in the SAN that must be addressed before the metro node is impacted. These problems may, or may not, be specific to the metro node. Sometimes, it may be required to disable the problem ports that are identified by the script as having or seeing a problem until the SAN problem is located and addressed.

The FC port monitoring script has the following features:

- Polls only metro node FC ports once a minute and will send an email to a configured email address if it detects a possible fabric issue.
- Explicitly identifies the cluster, director, and port experiencing the fabric issue.
- Reports on any degraded FC initiator-target pairs.
- The thresholds in the script can be modified in the json configuration file.
- Suppress error reports after 5 minutes, after which a summary email will be sent detailing the port error reports during the period where email were being suppressed.

NOTE: It is intended that support would work with the end user to deploy the monitoring script for setting up the port-stats-monitor script for the email server address and email list for those persons they want to receive the reports that the script will send out.

Example: port-monitor start [--email <email>,<email>,...]

Setting up the script for e-mailing of reports

Start the script and connect to the email (SMTP) server of the end user.

```
VPlexcli:/> port-monitor start --smtp <mail server ip address> -e [<email>,<email>,...]
```

NOTE: Once the script is started, then the output from it will be recorded in the file **ports-stats-monitor.log** which is viewable under `/var/log/VPlex/cli`.

Checking the script status

Steps

1. Check the status of the script to see if it is running.

```
VPlexcli:/> port-monitor status
    Status: running with the following parameters:
            Emails: joe@dell.com
            SMTP: x.x.x.x
            Local-only: False
```

2. To ensure the script restarts on the chance there is a reboot or restart of the management server, you can add persistence by adding the command that is used to start the script back in **Starting the script** to the `VPlex-init` file under the `/var/log/VPlex/cli directory` as shown in this step. Use the vi editor, and add the script start command line to end of `/var/log/VPlex/cli/VPlexcli-init` file.

```
Sample output:
service@ManagementServer:/var/log/VPlex/cli> vim VPlexcli-init

#-------------------------------------------------------------------------------
    #- (C) 2007-2010 EMC Corporation. All rights reserved.
    #-
    #- This CLI initialization script is executed if it's located in any of the
    #- following locations:
    #- (CLI terminates the search on first success.)
    #- if the --init-file option is specified on the command line then use that file
    #- else search for the file "VPlexcli-init" in the following order:
    #-   a.  CLI directory (specified with the --cli-directory option)
    #-   b.  current dir (of the shell that started CLI)
    #-   c.  user.dir (usually equivalent to the current dir)
    #-   d.  user.home
    #-   e.  classpath
    #- This script is processed as if it had been sourced using the 'source' command
    #-------------------------------------------------------------------------------
    .
    .
    ll /monitoring/directors/*/monitors/
    #
    #
         <new entry added below at the end of VPlex-init file,
     script -i port_stats_monitor
     port-monitor start -smtp <mail server ip address> -e <email>,<email>,...>
```

Adjust Thresholds (if needed)

Steps

On the management server, both if a Metro, create a directory `port-stats-monitor` and copy the specific hardware, VS2 or VS6, `config.json` file, you saw earlier after uncompressing the `port-stats-monitor_6.2.zip` file, to the newly created directory.

a. Create the directory `/var/log/VPlex/cli/port-stats-monitor`.

   Example: `mkdir /var/log/VPlex/cli/port-stats-monitor`

b. Copy the appropriate hardware `<vsX>_config.json` file to this directory for the metro node hardware you are working on. Examples: `cp vs2-config.json /var/log/VPlex/cli/port-stats-monitor/config.json` or `cp vs6-config.json /var/log/VPlex/cli/port-stats-monitor/config.json`.

   NOTE: For **Step c**, do not make any changes after loading the script. Let the monitor script run for a bit and if there are performance issues, then the end user will get email alerts of any issues and they will reach out to metro node support for further assistance. Go to **Step d**, though only to confirm that the monitor is running. In **Step d** scroll down to where

it reads "Checking status" and only run that command for now. Steps c and d are to be followed for both clusters if a Metro.

c. Changing default thresholds in the `config.json` file (optional). If you find that the default values, or one of them, could be increased for better results, you can modify the `config.json` file for new threshold values (using VI editor). Example: `vim /var/log/VPlex/cli/port-stats-monitor/config.json`.

```
Sample Output:
{
        "bad_CRC": 5,
        "Disc_frame": 40,
        "link_fail": 15,
        "Loss_of_sync": 45,
        "loss_of_sig": 45,
        "reset": 5
}
```

d. After doing modifications to the `config.json` file, you must restart the port-monitor script.

```
VPlexcli:/>  port-monitor restart
VPlexcli:/>  port-monitor status
Status: running with the following parameters:
         Emails: joe@dell.com <<< this will only show e-mail addresses if configured
         SMTP: x.x.x.x
         Local-only: False
         Threshold config: {u'lr-remote': 5, u'crc-errors': 50, u'invalid-transmission-
word': 500, u'link-failure': 10, u'loss-of-signal': 45, u'loss-of-sync': 60}
```

Port Stats Monitoring Usage Information

Usage: Taken from the 6.2.x script

```
Port Stats Monitoring

A prodscript for monitoring critical statistics for ports.

## What does this monitor do?

The monitor periodically logs VPLEX FC port statistics and can notify via email if
critical stats have increased past their threshold within a minute interval.

## Usage

After importing the prodscript with `script -i port_stats_monitor`, 5 commands are
created:

port-monitor restart     Restart all monitor threads.
port-monitor start       Start periodically monitoring for port stat changes
port-monitor status      Display the status of the port monitor thread
port-monitor stop        Stop any in-progress port stat monitor threads.
port-monitor test-email  Test the monitor's email notification.

### Starting the monitor

To start the monitor, run:
` port-monitor start [--email <email>,<email>...]`

options (* = required):
  -h | --help
          Displays the usage for this command.
  --verbose
          Provides more output during command execution.  This may not have any effect
for some commands.
  -e | --email= <emails> [, <emails> ...]
          Comma-separated email addresses to notify upon detecting a failure
  --smtp= <smtp>
          SMTP server address to use for notification emails
  --local-only
          Poll only cluster-local directors
```

```
VPlexcli:/> port-monitor start -e example@emc.com
Starting port stat monitor...

### Stopping the monitor

To stop the monitor, run `port-monitor stop`.


### Checking status

To see whether or not the monitor is running, or to see if any unexpected errors were
encountered, run the `port-monitor status` command:

VPlexcli:/> port-monitor status
Status: running with the following parameters:
        Emails: None
        SMTP: x.x.x.x
        Local-only: False
        Threshold config: None

### Restarting the monitor

If you wish to restart a stopped monitor with the same parameters as before, run
`port-monitor restart`. If you wish to use different options, use the `start` command
documented above.

## Configuring the driver-specific thresholds

The thresholds may be overridden by placing a JSON file at
/var/log/VPlex/cli/port-stats-monitor/config.json, with each key representing a stat to
monitor and the value representing the threshold at which to notify the user. Example
contents of the config.json:
{
  "crc-errors": 40,
  "link-failure": 15,
  "loss-of-sync": 45,
  "loss-of-signal": 45,
  "invalid-transmission-word": 40,
  "lr-remote": 5
}
```

Sample Output

Sample output of email that may be sent to the contact.

```
From: VPLEX Port Stat Notifier [mailto:vplex-port-stat-notifier@dell.com]
Sent: Day, Month date, YYYY H:MM <AM/PM>
To: <recipient>
Subject: VPlex Port Stat Notification for x.x.x.x <Serial Number>

The port stat monitor detected a problem.
Historical data is located in /var/log/VPlex/cli/port-stats-monitor.log

Current thresholds: crc-errors: 40, invalid-transmission-word: 40, link-failure: 15,
loss-of-signal: 45, loss-of-sync: 45

In the last 60 seconds:
director-1-1-A A1-FC03 (back-end) crc-errors has increased by 10924
director-1-1-A A1-FC02 (back-end) crc-errors has increased by 9541
director-1-1-A A1-FC01 (back-end) crc-errors has increased by 13655
director-1-1-A A1-FC00 (back-end) crc-errors has increased by 14982
The following I-Ts on director-1-1-A were banished:
        x fcp i 0xc00144878f0e0800 t 0x500601683660190e

The following additional reports from the last hour were suppressed:

2019-03-22 14:21:12
director-1-1-B B0-FC02 (front-end) crc-errors has increased by 13354
director-1-1-B B0-FC03 (front-end) crc-errors has increased by 19255
```

```
director-1-1-B B0-FC00 (front-end) crc-errors has increased by 15254
director-1-1-B B0-FC01 (front-end) crc-errors has increased by 953630
```

Things to note

Take note of the number of ports and the number of directors reporting issues. For instance, if half of the ports are reporting issues, then it may indicate a fabric-wide event. Whereas if only one port is reporting an error, then the problem is localized to a specific I-T Nexus.

The script is designed to suppress email after 5 minutes (as to not flood an email server). At that point, it will only report once an hour. The firmware connect to management server will contain all reports including any that were suppressed to email.

The following table contains a list of stats being monitored. What is being monitored depends on the hardware type, VS2 or VS6, and the GeoSynchrony code level. While the script can be applied to any code level at 6.0 SP1 (6.0.1.00.00.08) and above, what it can monitor depends the availability of the underlying stats. See below in the attachment (restricted) section a larger view of this table.

Brocade Switch Counter	Description	Threshold Rates	VS2 6.0 and newer	VS6 6.0 6.0.1 P7	VS6 6.1 and newer
crc err	Number of frames with CRC errors received (Rx)	45/min, error log, email, port fence; 5/min to unfence	None. But increments Disc_frame. Details in 13.5.10	None. Code never increments 'bad-crc' and 'bad_CRC'	crc-errors. Details in 2.7.5.1, 3.4.17. bad Eof will also increase
crc g_eof	Number of frames with CRC errors with good EOF received (Rx). [JS] This is fairly unique to Brocade but if we have the ability to determine if the CRC has good end of frame we could tell if the CRC was on the VPLEX link or further downstream		bad_CRC. Details in 13.5.10	None	crc-errors. Details in 3.4.17
bad eof	Number of frames with bad end-of-frame delimiters received (Rx).		Not everything RX_EOFa. Details in 13.5.12	None. now-EOFa code never increments this	Sum of nx-EOFa, ix-EOFdt, ix-EOFni, ni-SOFf. See 3.4.17
link fail	Number of link failures (LF1 or LF2 states) received (Rx).	Link loss. 15/min, error log, snmp trap	link_fail. Details in 8.7.4	none	link-failure. 3.4.17
loss sync	Number of times synchronization was lost (Rx).	45/min, Error log, snmp trap.	Loss_of_sync. Details in 8.7.4	none	loss-of-sync. 3.4.17
loss sig	Number of times a loss of signal was received (increments whenever an SFP is removed) (Rx).	45/min, Error log, snmp trap.	loss_of_sig. Details in 8.7.4	none	loss-of-signal. 3.4.17
Invalid Trasmission word	The number of times an invalid transmission word error occurs on a port. A word did not transmit successfully, resulting in encoding errors. Invalid word messages usually indicate a hardware problem.	45/min, error log, snmp trap, port fence 25/min to unfence	None.	none	invalid-transmission-word
I-Ts Marked Degraded/Undegraded over the interval	The number if I-Ts a director port has marked as degraded or undegraded over the time interval. A degraded I-T is not used to service customer I/O	Any change	Banished and Unbanished I-Ts	Banished and Unbanished I-Ts	Banished and Unbanished I-Ts

Logging: The logging file `port-stats-monitor.log` can be found on the management server in the `/var/log/VPlex/cli/ directory`.This log file is gathering raw data. The grep command [`grep "back-end\|front-end\|wan-com" /var/log/VPlex/cli/port-stats-monitor.log`] can produce a summary that is related to error reported in the `port-stats-monitor.log` file.

```
Example:
grep "back-end\|front-end\|wan-com" /var/log/VPlex/cli/port-stats-monitor.log

/var/log/VPlex/cli/port-stats-monitor.log.9:director-1-1-B B1-FC02 (back-end) invalid-
transmission-word has increased by 2956
/var/log/VPlex/cli/port-stats-monitor.log.9:director-1-1-B B1-FC02 (back-end) loss-of-
sync has increased by 443
/var/log/VPlex/cli/port-stats-monitor.log.9:director-1-1-B B1-FC02 (back-end) invalid-
transmission-word has increased by 3494
/var/log/VPlex/cli/port-stats-monitor.log.9:director-1-1-B B1-FC02 (back-end) loss-of-
sync has increased by 528
/var/log/VPlex/cli/port-stats-monitor.log.9:director-1-1-B B1-FC02 (back-end) invalid-
transmission-word has increased by 5996
```

Statistics

Metro node collects and reports three types of statistics:

- *counters* - Monotonically increasing value (analogous to a car's odometer)

- Counters are used to count bytes, operations, and errors.
- Often reported as a rate such as counts/second or KB/second.

- *readings* - Instantaneous value (analogous to a car's speedometer)
  - Readings are used to display CPU utilization, memory utilization.
  - Value can change every sample.
- *period-average* - Average of a series calculated over the last sample period. If:
  - *current_reading_sum* is the sum of all readings for the particular statistic since the monitor's creation.
  - *previous_reading_sum* is the count of all readings for the statistic since the monitor's creation.
  - *period-average* = (*current_reading_sum* - *previous_reading_sum*) / (*current_reading_count* - *previous_reading_count*)

Many statistics require a target port or volume to be specified. Output of the `monitor stat-list` command identifies which statistics need a target defined, and the type of target required when a monitor is created.

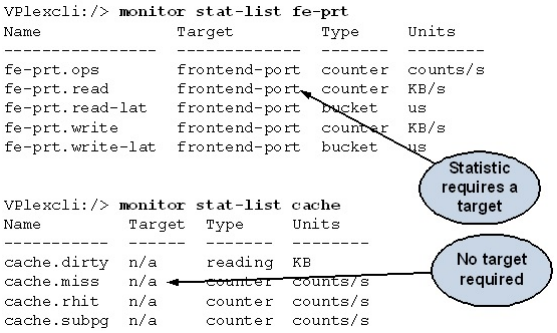

Figure 10. Monitoring targets

Display available statistics

Statistics are grouped into sub-categories.

Use the `monitor stat-list` command followed by the <**Tab**> key to display the statistics sub-categories. For example:

```
VPlexcli:/> monitor stat-list be-prt, cache, cg, director, directory, fc-com-port, fe-director, fe-lu, fe-prt, ip-com-port, ramf, rdma, storage-volume, virtual-volume, wrt-pacing
```

Use the `--categories categories` option to display the statistics available in the specified category. For example:

```
VPlexcli:/monitoring> monitor stat-list --categories director
Name                    Target  Type     Units
----------------------  ------  -------  --------
director.be-aborts      n/a     counter  counts/s
director.be-ops         n/a     counter  counts/s
director.be-ops-read    n/a     counter  counts/s
director.be-ops-write   n/a     counter  counts/s
director.be-read        n/a     counter  KB/s
.
.
.
```

Use the * wildcard to display all statistics for all categories.

For example:

```
VPlexcli:/> monitor stat-list *
Name                                            Target             Type     Units
----------------------------------------------- -----------------  -------  --------
be-prt.read                                     backend-port       counter  KB/s
be-prt.write                                    backend-port       counter  KB/s
cache.dirty                                     n/a                reading  KB
cache.miss                                      n/a                counter  counts/s
cache.rhit                                      n/a                counter  counts/s
cache.subpg                                     n/a                counter  counts/s
cg.closure                                      consistency-group  bucket   us
cg.delta-util                                   consistency-group  reading  %
```

```
cg.drain-lat                                    consistency-group  bucket   us
cg.exch-bytes                                   consistency-group  counter  KB/s
cg.exch-lat                                     consistency-group  bucket   us
cg.exch-pages                                   consistency-group  counter  counts/s
cg.input-bytes                                  consistency-group  counter  KB/s
cg.input-ops                                    consistency-group  counter  counts/s
cg.inter-closure                                consistency-group  bucket   us
cg.outOfDate-counter                            consistency-group  counter  counts/s
cg.pipe-util                                    consistency-group  reading  %
cg.write-bytes                                  consistency-group  counter  KB/s
cg.write-lat                                    consistency-group  bucket   us
cg.write-pages                                  consistency-group  counter  counts/s
.
.
.
```

Front-end performance statistics

Metro node collects detailed performance statistics on its virtual volumes, which primarily includes the Read and Write statistics with the I/O size and the LBA information. You can use this data to identify and resolve any I/O performance issues with metro node.

This feature is enabled in metro node by default. The collected statistics is available in the `fe_perf_stats_<timestamp>.log` file at the `/var/log/VPlex/cli/` folder. The file includes these details:

Table 13. Front-end performance statistics

Field	Description
vol	Name of the virtual volume
Enabled queue	Name of the queue
pos	Serial number of the task in the queue
I	WWN of Initiator port
T	WWN of target port
status	Internal status or the Cache status
time	The time for which the I/O task has been running (In `usec`)
opcode	Operation code of the command (if applicable)
LBA	Value of the logical block addressing (LBA) element in the command (if applicable)
len	Blocks or bytes that are being transferred or verified (if applicable)

To manage the performance the front-end statistics collection, use these commands at any metro node CLI context:

- `front-end-performance-stats stop` - stops a running performance statistics collection.
- `front-end-performance-stats start` - starts a running performance statistics collection.
- `front-end-performance-stats status` - displays the status of front-end performance statistics collection.

NOTE: For more information on the commands, see the *CLI Reference Guide for metro node*.

Statistics tables

The following tables list the statistics in each category:

- Back-end fibre channel port (be-prt) statistics
- Cache statistics
- Director statistics
- Front-end director (fe-director) statistics

- Front-end volume (fe-lu) statistics
- Front-end port (fe-prt) statistics
- Remote RAID (ramf) statisticsRemote RAID (ramf) statistics
- Storage-volume statistics
- Virtual-volume statistics
- IP WAN COM (ip-com-port) statistics - Monitors IP ports (any port with GE or XG in the port name).
- IP Congestion Control Statistics
- COM cluster I/O statistics
- COM Path statistics
- COM Endpoint Statistics
- XCOPY Statistics
- Host Initiator Statistics

Table 14. Back-end fibre channel port (be-prt) statistics

Statistic	Type	Description
be-prt.read type: counter, units: bytes/second, arguments: port#	Back-end port reads	Number of bytes read through the specified FC port.
be-prt.write type: counter, units: bytes/second, arguments: port#	Back-end port writes	Number of bytes written through the specified FC port.

Table 15. Director statistics

Statistic	Type	Description
director.async-write	Back-end writes	Number of asynchronous writes in KB/second.
director.be-aborts type: counter, units: counts/second, arguments: none	Back-end operations	Number of aborted I/O operations through the director's back-end ports.
director.be-busies	back-end operations	Number of busy notifications on this director.
director.be-ops type: counter, units: counts/second, arguments: none	Back-end operations	Number of I/O operations through the director's back-end ports.
director.be-ops-read type: counter, units: counts/second, arguments: none	Back-end reads	Number of reads by the director's back-end ports.
director.be-ops-write type: counter, units: counts/second, arguments: none	Back-end writes	Number of writes through the director's back-end ports.
director.be-ops-ws	Back-end operations	Number of back-end write same operations
director.be-qfulls	back end writes	Number of queue full notifications for this back end port.
director.be-read type: counter, units: bytes/second, arguments: none	Back-end reads	Number of bytes read by the director's back-end ports.
director.be-resets	counter	Number of back end resets per second
director.be-timeouts	counter	Number of back end timeouts per second.
director.be-unitattns	counter	Number of back end unit attentions per second.

Table 15. Director statistics (continued)

Statistic	Type	Description
director.be-write type: counter, units: bytes/second, arguments: none	Back-end writes	Number of bytes written by the director's back-end ports.
director.be-ws type: counter, units: bytes/second, arguments: none	Back-end WriteSame	Back-end WriteSame details.
director.busy type: reading; units: percentage, arguments: none	CPU	Percentage of CPU usage.
director.com-bytes-active type: reading, units: counts, arguments: target director	Communication bytes active	Number of bytes active to a remote director.
director.com-bytes-queued type: reading, units: counts, arguments: target director	Communication bytes queued	Number of bytes queued to a remote director.
director.com-ops-active type: reading, units: counts, arguments: target director	Communication operations active	Number of operations active to a remote director.
director.com-ops-queued type: reading, units: counts, arguments: target director	Communication operations queued	Number of operations queued to a remote director.
director.dr1-rbld-recv type: counter, units: bytes/second, arguments: none	Rebuild bytes received	Number of bytes received by this node from remote node(s) for rebuild traffic (reads and/or writes).
director.dr1-rbld-sent type: counter, units: bytes/seconds, arguments: none	Rebuild bytes sent	Number of bytes sent by this node to remote node(s) for rebuild traffic (reads and/or writes)
director.fe-ops type: counter, units: counts/second, arguments: none	Front-end operations	Number of I/O operations through the director's front-end ports.
director.fe-ops-act type: reading, units: counts, arguments: none	Front-end operations active	Number of active outstanding I/O operations on the director's front-end ports.
director.fe-ops-q type: reading, units: counts, arguments: none	Front-end operations queued	Number of queued outstanding I/O operations on the director's front-end ports.
director.fe-ops-read type: counter, units: counts/second, arguments: none	Front-end reads	Number of reads on the director's front-end ports.
director.fe-ops-write type: counter, units: counts/second arguments: none	Front-end writes	Number of writes on the director's front-end ports.

Table 15. Director statistics (continued)

Statistic	Type	Description
director.fe-read type: counter, units: bytes/second, arguments: none	Front-end reads	Number of bytes read from the director's front-end ports.
director.fe-write type: counter, units: bytes/second, arguments: none	Front-end writes	Number of bytes written to the director's front-end ports.
director.heap-used type: reading; units: percentage, arguments: none	Memory	Percentage of memory usage on the director.
director.per-cpu-busy type: reading, units: percentage, arguments: none	CPU busy	The total utilization (user and system) of each CPU in the director.
director.msg-send-ops	number of operations	The total number of messages sent from this director.
director.msg-max-lat	Maximum latency	The maximum latency of messages sent from this director.
director.msg-min-lat	Minimum latency	The minimum latency of messages sent from this director.
director.msg-avg-lat	Average latency	The agerage latency of messages sent from this director.

Table 16. Front-end director (fe-director) statistics

Statistic	Type	Description
fe-director.aborts type: counter, units: counts/second, arguments: none	Front-end operations	Number of aborted I/O operations through the director's front-end ports.
fe-director.caw-lat type: bucket, units: microsecond, arguments: none	CompareAndWrite operations latency	CompareAndWrite latency in microseconds on the specified director's front-end ports. The latency bucket is reduced to three buckets from 0 to maximum instead of 64 latency buckets collected within the metro node firmware.
fe-director.read-lat type: bucket, units: microsecond, arguments: none	Front-end director read latency	Read latency distribution in microseconds on the director's front-end ports.
fe-director.write-lat type: bucket, units: microsecond, arguments: none	Front-end director write latency	Write latency distribution in microseconds on the director's front-end ports.
fe-director.ws16-avg-lat type: period-average, units: us, arguments: none	Front-end director writesame average latency	Average WriteSame latency distribution on the director's front-end ports.
fe-director.unmap-ops type: counter, units: counts/second, arguments: none	Front-end director unmap operations	Number of unmap operations per second on the specified front-end director
fe-director.unmap-avg-lat	Front-end director average unmap latency	Average latency in microseconds of unmap operations at the specified front-end director

Table 16. Front-end director (fe-director) statistics (continued)

Statistic	Type	Description
type: period-average, units:us, arguments: none		

Table 17. Front-end volume (fe-lu) statistics

Statistic	Type	Description
fe-lu.caw-lat type: bucket, units: microsecond, arguments:volume-id	CompareAndWrite operations latency	CompareAndWrite latency in microseconds on the specified front-end volume.
fe-lu.caw-mis type: counter, units: counts/second, arguments: volume-id	CompareAndWrite miscompares	Number of CompareAndWrite miscompares on the specified front-end volume.
fe-lu.caw-ops type: counter, units: counts/second, arguments: volume-id	CompareAndWrite operations	Number of CompareAndWrite operations on the specified front-end volume.
fe-lu.ops type: counter, units: counts/second, arguments: volume-id	Front-end volume operations	Number of I/O operations on the specified front-end volume.
fe-lu.read type: counter, units: bytes/second, arguments: volume-id	Front-end volume reads	Number of reads on the specified front-end volume.
fe-lu.read-lat type: bucket, units: microsecond, arguments: volume-id	Front-end volume read latency	Read latency distribution in microseconds on the specified front-end volume.
fe-lu.write type: counter, units: bytes/second, arguments: volume-id	Front-end volume writes	Number of writes on the specified front-end volume.
fe-lu.write-lat type: bucket, units: microsecond, arguments: volume-id	Front-end volume write latency	Write latency distribution in microseconds on the specified front-end volume.
fe-lu.ws16-avg-lat type: period-average, units: us, arguments: virtual-volume	Front-end volume average WriteSame latency	Average WriteSame latency distribution on the specified front-end volume.
fe-lu.ws16-ops type:counter, units: counts/second, arguments: virtual-volume	Front-end volume WriteSame operations	Number of WriteSame operations on the specified front-end volume.
fe-lu.unmap-ops type: counter, units: counts/second, arguments: virtual-volume	Front-end volume unmap operations	Number of unmap operations per second on the specified front-end volume
fe-lu.unmap-avg-lat type: period-average, units:us, arguments: virtual-volume	Front-end volume average unmap latency	Average latency in microseconds of unmap operations at the specified front-end volume

Table 18. Front-end port (fe-prt) statistics

Statistic	Type	Description
fe-prt.caw-lat type: bucket, units: microsecond, arguments:port#	CompareAndWrite operations latency	CompareAndWrite latency in microseconds on the specified front-end port.
fe-prt.caw-mis type: counter, units: counts/sec, arguments: port#	CompareAndWrite miscompares	Number of CompareAndWrite miscompares on the specified front-end port.
fe-prt.caw-ops type: counter, units: counts/sec, arguments: port#	CompareAndWrite operations	Number of CompareAndWrite operations on the specified front-end port.
fe-prt.ops type: counter, units: counts/sec, arguments: port#	Front-end port operations	Number of I/O operations on the specified front-end FC port.
fe-prt.read type: counter, units: bytes/sec, arguments: port#	Front-end port reads	Number of bytes read from the specified front-end FC port.
fe-prt.read-lat type: bucket, units: microsecond, arguments: port#	Front-end port read latency	Read latency distribution in microseconds on the specified front-end FC port.
fe-prt.write type: counter, units: bytes/second, arguments: port#	Front-end port writes	Number of bytes written to the specified front-end FC port.
fe-prt.write-lat type: bucket, units: microsecond, arguments: port#	Front-end port write latency	Write latency distribution in microseconds on the specified front-end FC port.
fe-prt.ws16-avg-lat type: period-average, units: us, arguments: frontend-port	Front-end port average WriteSame latency	Average WriteSame latency distribution on the specified front-end FC port.
fe-prt.ws16-ops type: counter, units: counts/second, arguments: frontend-port	Front-end port WriteSame operations	Number of WriteSame operations on the specified front-end FC port.
fe-prt.unmap-ops type: counter, units: counts/second, arguments: frontend-port	Front-end port unmap operations	Number of unmap operations per second seen at specified port.
fe-lu.unmap-avg-lat type: period-average, units:us, arguments: frontend-port	Front-end port average unmap latency	Average latency in microseconds of unmap operations at the specified front-end port.

Table 19. Remote RAID (ramf) statistics

Statistic	Type	Description
ramf.cur-op type: reading, units: counts, arguments: none	Current op count	Instantaneous count of remote RAID operations.

Table 19. Remote RAID (ramf) statistics (continued)

Statistic	Type	Description
ramf.exp-op type: counter, units: counts/second, arguments: none	Remote operations	Total number of remote IOPS.
ramf.exp-rd type: counter, units: bytes/second, arguments: none	Remote reads	Remote reads from another cluster to a disk or LUN at the local cluster.
ramf.exp-wr type: counter, units: bytes/second, arguments: none	Remote writes	Remote writes from another cluster to a disk or LUN at the local cluster.
ramf.imp-op type: counter, units: counts/second, arguments: none	Imported ops	Number of operations that have been requested by a given director, regardless of remote target.
ramf.imp-rd type: counter, units: bytes/second, arguments: none	Imported reads	Reads from the local cluster to a disk or LUN at a remote cluster.
ramf.imp-wr type: counter, units: bytes/second, arguments: none	Imported writes	Writes from the local cluster to a disk or LUN at a remote cluster.
ramf.imp-rd-avg-lat type: period-average, units: microseconds, arguments: none	Imported reads	Average latency of remote reads from the local cluster to a disk or LUN at a remote cluster.
ramf.imp-wr-avg-lat type: period-average, units: microseconds, arguments: none	Imported writes	Average latency of remote writes from the local cluster to a disk or LUN at a remote cluster.

Table 20. Storage-volume statistics

Statistic	Type	Description
storage-volume.per-storage-volume-read-latency type: bucket, units: microsecond, arguments: volume-id	Volume read latency	Read latency distribution in microseconds on the specified storage volume.
storage-volume.per-storage-volume-write-latency type: bucket, units: microsecond, arguments: volume-id	Volume write latency	Write latency distribution in microseconds on the specified storage volume.
storage-volume.read-latency type: bucket, units: microsecond, arguments: none	Average volume read latency	Average read latency distribution in microseconds on all storage volumes.
storage-volume.write-latency type: bucket, units: microsecond, arguments: none	Average volume write latency	Average write latency distribution in microseconds on all storage volumes.
storage-volume.write-same-avg-lat	Average volume WriteSame latency	Average writesame latency distribution on all storage volumes.

Table 20. Storage-volume statistics (continued)

Statistic	Type	Description
type: period-average, units: us, arguments: none		

Table 21. Virtual-volume statistics

Statistic	Type	Description
virtual-volume.dirty type: reading, units: counts, arguments: volume-id	Volume dirty	Number of modified pages in cache for the specified virtual volume.
virtual-volume.ops type: counter, units: counts/second, arguments: volume-id	Volume operations	Total number of I/O operations for the specified virtual volume.
virtual-volume.read type: counter, units: bytes/second, arguments: volume-id	Volume reads	Number of reads in bytes for the specified virtual volume.
virtual-volume.write type: counter, units: bytes/second, arguments: volume-id	Volume writes	Number of writes in bytes for the specified virtual volume.

Table 22. IP WAN COM (ip-com-port) statistics

Statistic	Type	Description
ip-com-port.recv-pckts	Counter, units: counts/second, arguments: port-name	Number of packets received through UDP on this IP WAN COM port.
ip-com-port.send-bytes	Counter, units: bytes/second, arguments: port-name	Number of bytes sent through UDP on this IP WAN COM port.
ip-com-port.send-drops	Counter, units: counts/second, arguments: port-name	Number of sent packets dropped on this IP WAN COM port.
ip-com-port.send-pckts	Counter, units: counts/second, arguments: port-name	Number of packets sent through UDP on this IP WAN COM port.
ip-com-port.recv-errors	IP WAN COM Port receive errors	Number receive errors on this WAN COM Port
ip-com-port.send-errors	IP WAN COM Port send errors	Number of send errors on this IP WAN COM Port
ip-com-port.recv-dropped	IP WAN COM Port received packets dropped	Number of received packets dropped on this IP WAN COM Port
ip-com-port.send-dropped	IP WAN COM Port sent packets dropped	Number of sent packets dropped on IP WAN COM Port
ip-com-port.recv-overruns	IP WAN COM Port receive overruns	Number of receive overruns IP WAN COM Port
ip-com-port.send-overruns	IP WAN COM Port send overruns	Number of send overruns on this IP WAN COM Port

Table 22. IP WAN COM (ip-com-port) statistics (continued)

Statistic	Type	Description
ip-com-port.recv-frame-errors	IP WAN COM Port received frames	Number of frames received on this IP WAN COM Port
ip-com-port.send-carrier-errors	IP WAN COM Port send carrier	Number of carriers sent on this IP WAN COM Port ?
ip-com-port.collisions	IP WAN COM Port collisions	Number of collisions on this IP WAN COM Port

Table 23. IP Congestion Control Statistics

Statistic	Description
ip-congestion-control.ip-wan-cc-rtt	Round trip time maintained by TCP in micro-seconds.
ip-congestion-control.ip-wan-cc-rttvar	RTT smoothed mean deviation maximum measured in microseconds.
ip-congestion-control.ip-wan-recv-bytes	Total number of bytes received on the TCPCOM path.
ip-congestion-control.ip-wan-recv-cnt	Total number of packets received on the TCPCOM path.
ip-congestion-control.ip-wan-retx-cnt	Total number of TCP retransmissions.
ip-congestion-control.ip-wan-send-bytes	Total number of bytes sent on the TCPCOM path.
ip-congestion-control.ip-wan-send-cnt	Total number of packets sent on the TCPCOM path.

Table 24. COM cluster I/O statistics

Statistic	Description
com-cluster-io.avg-lat type: reading, units: microseconds, arguments: cluster-id	Average latency in microseconds of all I/O from the local cluster to the other cluster in the last query period. Takes a cluster number as an argument
com-cluster-io.max-lat type:reading, units: microseconds, arguments: cluster-id	Maximum latency in microseconds of all I/O from the local cluster to the other cluster. Takes a cluster number as an argument.
com-cluster-io.min-lat reading, units: microseconds, arguments: cluster-id	Minimum latency in microseconds of all I/O from the local cluster to the other cluster. Takes a cluster number as an argument.
com-cluster-io.send-ops type:reading, units: none, arguments: cluster-id	Number of I/O send operations to the cluster.
com-cluster-io.ops-active	Current outstanding messages to a site.
com-cluster-io.bytes-active	Current outstanding bytes to a site.
com-cluster-io.bytes-queued	Current queued bytes to a site.
com-cluster-io.ops-queued	Current queued messages to a site.

Table 25. COM I/O group statistics

Statistic	Description
com-io-group.io-tm-avg	Average latency on this channel group in last 5 seconds (updated every 5 seconds).
com-io-group.io-tm-cnt	Messages sent on this channel group in last 5 seconds (updated every 5 seconds).
com-io-group.io-tm-max	Maximum latency on this channel group in last 5 seconds (updated every 5 seconds).

Table 25. COM I/O group statistics (continued)

Statistic	Description
com-io-group.io-tm-min	Minimum latency on this channel group in last 5 seconds (updated every 5 seconds).
com-io-group.msg-b-in	Always returns zero.
com-io-group.msg-b-out	Total bytes sent on this channel group.
com-io-group.msg-cnt-in	Always returns zero.
com-io-group.msg-cnt-out	Total messages sent on this channel group.

Table 26. COM Path Statistics

Statistic	Description
com-path.ping-count	Number of ping packets sent. These are used to help calculate latency.
com-path.ping-late	Number of ping packets that took too long.
com-path.ping-lost	Number of ping packets lost.
com-path.posted-bytes	Number of transmit bytes posted. (Bytes queued for transmission).
com-path.posted-send-ack	Number of ACK buffers posted. (ACK buffers queued for transmission).
com-path.posted-send-ctl	Number of control buffers posted. (Control buffers queued for transmission).
com-path.rtt-avg	Average round trip time for data to travel along the path.
com-path.rtt-max	Max round trip time for data to travel along the path.
com-path.rtt-min	Min round trip time for data to travel along the path.
com-path.send-bytes	Number of data bytes send along this path. This includes data plus UDCOM headers.
com-path.send-posted-bytes	Number of transmit data buffers posted. Put another way, data queued for transmission.

Table 27. COM Endpoint Statistics

Statistic	Description
com-endpoint.ack-bytes-recv	Number of ACK bytes received.
com-endpoint.ack-bytes-sent	Number of ACK bytes sent.
com-endpoint.ack-pckts-recv	Number of ACK packets received.
com-endpoint.ack-pckts-sent	Number of ACK packets sent.
com-endpoint.cx-bad-ver	Number of incorrect version of control packets.
com-endpoint.cx-bytes-recv	Number of control bytes received.
com-endpoint.cx-bytes-sent	Number of control bytes sent.
com-endpoint.cx-pckts-recv	Number of control packets received.
com-endpoint.cx-pckts-routed	Number of routed control packets.
com-endpoint.cx-pckts-sent	Number of control packets sent.
com-endpoint.data-bytes-recv	Number of data bytes received.
com-endpoint.data-bytes-sent	Number of data bytes sent.
com-endpoint.data-padding-recv	Number of padding data packets received.
com-endpoint.data-pckts-badkey	Number of packets that have an invalid domain key.

Table 27. COM Endpoint Statistics (continued)

com-endpoint.data-pckts-badlen	Number of invalid data packet length.
com-endpoint.data-pckts-recv	Number of data packets received.
com-endpoint.data-pckts-routed	Number of routed data packets.
com-endpoint.data-pckts-runt	Number of data packets that are less than 64 bytes long.
com-endpoint.data-pckts-sent	Number of data packets sent.
com-endpoint.rx-ack-buf-pend-pckts	Number of pending ack buffers to be processed. This is the number of ACK packets that have come in, but not yet processed.
com-endpoint.rx-credits	Number of receive credits.
com-endpoint.tx-posted-bytes	Number of transmitted bytes posted. (Bytes queued to be transmitted).

Table 28. XCOPY Statistics

Statistic	Description
fe-director.xcopy-avg-lat	Average latency to process all front-end received XCOPY for a given director, in micro seconds. Automatically collected as a part of perpetual monitoring. The collected values are available through the perpetual monitoring file localed on metro node Management Server at /var/log/VPlex/cli/director-[1|2]-[1|2]-[A|B]_PERPETUAL_vplex_sys_perf_mon.log
fe-director.xcopy-ops	The number of XCOPY operations completed per second for a given director.
fe-lu.xcopy-avg-lat	Average latency to process front-end XCOPYreceived for a given metro node virtual volume, of a specific director,in micro seconds, for a given metro node virtual volume
fe-lu.xcopy-ops	Count of XCOPY operations processed by a given metro node virtual volume of a specific director
fe-prt.xcopy-avg-lat	Average latency to process front-end XCOPY received on a given front-end port, of a specific director, in micro seconds at the port level
fe-prt.xcopy-ops	Count of XCOPY operations processed by a given metro node front-end port of a specific director

Table 29. Host Initiator Statistics

Statistic	Description
host-init.unmap-ops type: counter, units: counts/second, arguments: none	Host initiator unmap operations.
host-init.unmap-avg-lat type: period-average, units:us, arguments: none	Host initiator average unmap latency.

A

Metro node with active-passive storage arrays

Topics:

Active-passive array

An active-passive array typically has two controllers and provides active-passive access to a Logical Unit (LU) through a set of target ports. The access types of these ports are Active (ACT) or Passive (PAS). Active is used for I/O and passive cannot be used for I/O. When active paths to logical units are lost, the initiator (metro node) can decide to activate passive paths to perform I/Os by sending vendor specific SCSI commands to the array.

The controller with the active target ports for a specific logical unit is referred to as Active (ACT) controller of that logical unit. The controller with passive target ports for a specific logical unit is referred to as Passive (PAS) controller of that logical unit. The controller which is active for a logical unit can be a passive controller for some other logical unit and vice versa.

ALUA mode enabled array

An Asymmetric Logical Unit Access (ALUA) mode enabled storage array provides active/active access to a logical unit through all the target ports. Based on their bandwidth, these ports are sorted into preferred and non-preferred Target Port Groups (TPG). The higher bandwidth preferred target ports have access state of Active/Active Optimized (AAO) while non-preferred target ports have access state of Active/Non-Optimized (AAN). In the absence of AAO paths, I/Os continue on AAN paths.

The controller with preferred target ports for a specific logical unit, is referred to as Active/Active-Optimized (AAO) controller of that logical unit while the controller with non-preferred target ports for a specific logical unit, is referred to as Active/Non-Optimized (AAN) controller of that logical unit. The controller which is AAO for a logical unit can be a AAN controller for some other logical unit and vice versa. For the purpose of ALUA enabled logical unit failover processing, ALUA access state of Active/Active-Optimized (AAO) equates to an Active (ACT) path and Active/Active-NonOptimized (AAN) equates to a Passive (PAS) path, internally.

Targets advertise their support for ALUA on a per logical unit basis through a standard inquiry response. There are three different modes of operation :

Implicit ALUA The target device can independently change the logical unit access states internally.

Explicit ALUA The target device requires an initiator to change the logical unit access states by sending specific SCSI commands, when necessary.

Implicit-Explicit ALUA Has the advantage of both implicit and explicit ALUA. Targets might support implicit ALUA, explicit ALUA, or implicit-explicit ALUA.

Logical Unit failover execution

When the logical unit is accessible through all the paths, the active controller becomes the preferred controller, and when no active paths are available, the passive controller becomes the preferred controller. The failover of the logical unit is triggered by the master director in the metro node cluster, when the preferred controller is not its active controller. The master director in the cluster initiates the logical unit failover by sending vendor-specific SCSI commands to the target device to change the access state of the logical unit. Based on the response received from the target device for the command, the logical unit failover either succeeds or fails.

When failover is initiated for a specific logical unit on an array to a specific target controller as active, metro node firmware event apf/3 is observed. When failover succeeds or fails for a specific logical unit on an array to a specific target controller as active, a metro node firmware event apf/4 is generated.

For Example:

```
apf/3 Failover initiated for logical unit VPD83T3:6006016015a0320061d7f2b300d3e211 on
array EMC~CLARiiON~FNM00124500474 to target controller FNM00124500474.SPA as active.
```

```
apf/4 Failover succeeded for logical unit VPD83T3:6006016015a0320061d7f2b300d3e211 on
array EMC~CLARiiON~FNM00124500474 to target controller FNM00124500474.SPA as active.
```

```
apf/4 Failover failed for logical unit VPD83T3:600601606bb72200f01fb4fa1e22e311 on array
EMC~CLARiiON~FCNCH072602809 to target controller FCNCH072602809.SPA as active. reason:
Scsi mode select command failed
```

Similar entries can be found in the metro node firmware event log `/var/log/VPlex/cli/firmware.log*` on a running management server.

Logical Unit failback

When the status of the logical unit becomes nominal, metro node automatically attempts to fail back the logical unit to its default controller. This is usually defined as the owner of the logical unit as determined by the array. The logical unit failover execution process is started again in order to optimize the performance on the storage array side. This failback occurs only if the array has the `autoswitch` property enabled and the logical unit is visible through the controller.

NOTE: The *Simple Support Matrix for metro node* provides more information on the supported Dell EMC storage systems and third-party arrays.

Index

R

S

T

U

V

W