Performance analysis of the HP StorageWorks Enterprise Virtual Array storage systems using HP StorageWorks Command View EVAPerf

Introduction

Scope

This document provides guidelines and techniques for analyzing the data provided by HP StorageWorks Command View EVAPerf and detailed information on the data it presents. It is not intended to be an in-depth performance troubleshooting or storage array-tuning manual. The document does not cover any information required to install, configure, or execute the tool.

Target audience

This document is intended for storage administrators or architects involved in performance characterization of an HP StorageWorks Enterprise Virtual Array (EVA). You should have a basic understanding of storage performance concepts, as well as EVA architecture and management. It is assumed that you are familiar with the operation and output selections available in HP Command View EVAPerf.

Additional resources

- For information on acquisition, installation, and execution of the HP StorageWorks Command View EVAPerf tool, refer to the *HP StorageWorks Command View EVA User Guide* included on the HP StorageWorks Command View EVA distribution CD.
- For information on the EVA models refer to the following websites:

  http://www.hp.com/go/eva3000

  http://www.hp.com/go/eva5000

  http://www.hp.com/go/eva4000

  http://www.hp.com/go/eva6000

  http://www.hp.com/go/eva8000

- For information on HP Command View EVAPerf and related documentation, refer to the HP StorageWorks Command View EVA website at http://h18006.www1.hp.com/products/storage/software/cmdvieweva/index.html.

HP Command View EVAPerf overview

Many performance monitoring tools and utilities are available for host (server) and storage array network (SAN) infrastructure. These tools measure the workload that the host presents to an attached EVA and provide an indication of the ability of the array to process the load. For example, host-based parameters such as request rates, data rates, and latencies can provide a good picture of the activity on the array. However, a server only reports on its own activity against any given virtual volumes that it accesses. If a composite picture of the activity workload to EVA is required, then HP Command View EVAPerf is required.

HP Command View EVAPerf also provides information on activity within the storage system that can be useful in determining saturation levels or assessing capacity for additional workload. Measurements reported at host port and physical disk levels can, for example, expose possible bottlenecks within the array. This analysis is similar to looking into the discrete components within a computer system—CPU, memory, paging file—to identify bottlenecks within the host.

While HP Command View EVAPerf provides a substantial amount of performance and associated information for several components within the storage array, it cannot discern other important array configuration parameters, which can be critical in understanding the collected data. For an in-depth performance investigation, it might be necessary to acquire additional information not directly available from HP Command View EVAPerf, such as:

- Virtual disk RAID (Vraid) levels
- Snapshot, Snapclone, and HP Continuous Access EVA remote mirror volume identification
- Disk group configuration and physical layout
- Virtual and physical disk sizes or capacities
- Disk group occupancy

Additional information that might be helpful for documentation and problem solving includes:

- Host bus adapter (HBA) firmware and driver revisions
- All driver and registry parameters
- SAN fabric map and switch firmware levels
- EVA firmware
- Application workload profiles

Capturing performance information with HP Command View EVAPerf

HP Command View EVAPerf provides many parameter choices for capturing workload and performance information from the EVAs visible to the tool. Command line execution is straightforward and enables you to display that information on the screen or save to a file for later review.

The first step is to define the intended use for the resulting data. Three decisions must be made regarding data capture: the duration of the capture, the frequency at which the counters will be sampled, and the selection of counters to be sampled. All choices depend on how the data is to be used.

Single sample

If you want to monitor the storage system interactively to determine the current state of the system, the capture duration, consisting of a single sample, is appropriate and effective. This method enables the user to quickly determine the workload, performance, and some configuration parameters for the system. Some examples where this method might be helpful are:

- Total load to each EVA system
- Controller processor utilization
- Virtual disk read hit percentage
- Disk group loading

These quick views provide a convenient snapshot of the system behavior at the moment of sample and enable you to determine where to focus for more in depth investigation.

When using single sample capture, it is usually most helpful to restrict the data to a single performance object, such as host port statistics, and simply display the results on the monitor screen. You have maximum flexibility in viewing different objects in quick succession to determine where the point of interest lies or deciding where more investigation must be done. Viewing all objects in a single sample can be difficult to view after capture and requires the longest time to execute.

Short-term duration

A longer view of system activity is helpful when investigating behavior that cannot fully be demonstrated with a single sample. Examples of such behavior are:

- Transient effects of workload
- Internal system behavior (such as physical disk activity)
- External factors such as inter-site link (ISL) behavior
- Problems which occur at specific times during the work day

In such cases, durations of 15 minutes to a few hours would be appropriate. Sample intervals, particularly for transient behaviors (for example, latency spikes) would then be as short as practically possible, keeping with timing constraints discussed in the following section.

Long-term duration

The longest durations of data capture would be used for background performance and workload logging (base lining), for capturing an infrequent event, or for multiple, long-term events. This duration ranges from hours to continuous, depending on the objective.

Sample intervals for this mode of operation would tend to be longer, ranging from tens of seconds to minutes per sample, again depending on the objective. Capturing short-term phenomena requires short sample intervals as described in the previous section. Long sample intervals on the HP StorageWorks Enterprise Virtual Array 4000 (EVA4000), HP StorageWorks Enterprise Virtual Array 6000 (EVA6000), and HP StorageWorks Enterprise Virtual Array 8000 (EVA8000) will show less variation because of averaging over the sample interval.

When generating an historical record of EVA workload and response, it is often desirable to limit the data to that which is most useful. Restricting the number of performance objects captured optimizes the sample rate, reduces the amount of storage required to hold the data, and reduces the effort to review the data later. To accomplish this, you must execute multiple instances of HP Command View EVAPerf, each capturing a single desired performance object. The most valuable information is captured from the controller statistics, host ports, virtual disks, and physical disk groups. Additional benefit can be derived from host connections and, if applicable, HP Continuous Access tunnels.

After selecting counters, the next decision involves how to select the sample rate and duration. If the information is to be used primarily for long-term characterization of workload, then sample intervals of one to five minutes should be adequate. If it is necessary to keep a detailed record of workload and performance for anticipated problem solving activities, then a finer granularity is required and 10 to 60 seconds is appropriate.

Unless otherwise specified, the duration is forever. However, for data manageability and selection for retrieval, consecutive iterations of one- or two-hour duration are desirable. This requires scripting to launch the HP Command View EVAPerf executable at those intervals and creating the corresponding time-stamped output files.

An example of multiple execution instances with 20-second sample intervals with two-hour duration follows:

```
evaperf cs –cont 20 -dur 7200 > cs<datetime>.txt
evaperf hps –cont 20 -dur 7200 > hps<datetime>.txt
evaperf vd –cont 20 -dur 7200 > vd<datetime>.txt
evaperf pdg –cont 20 -dur 7200 > pdg<datetime>.txt
```

where `<datetime>` is the timestamp set by the scripting execution. As written, these commands will be executed sequentially, which is a problem because concurrent data is usually required. It is therefore necessary to execute HP Command View EVAPerf in multiple, separate terminal sessions or use a background scheduling utility for concurrent execution of each command.

The plain text output format, which is the output from the preceding command examples, is excellent for readability. However, when post processing tools are used, it is best to capture in a tighter format such as a comma separated variable (CSV) file.

In the example, each execution samples at 20-second intervals and should therefore correlate to each other, although start times can be skewed. It is critical that all data captures complete in less time than the sample interval, or the slowest will run continuously and fall behind the others. For example, if there are 750 virtual disks over all systems sampled, it is unlikely the `vd` command execution shown in the preceding example will remain in lock-step with the others.

The larger the requested set of data, the longer it takes to capture it. Performance objects in systems that have many members, specifically virtual disks and physical disks, can take up to a minute to capture. Therefore, it is best to perform a brief test execution to determine the minimum setting of the collection interval before commencing a long-term data gathering session. Run HP Command View EVAPerf in continuous screen mode for a few samples. The amount of time required to capture the data in the sample is shown in milliseconds at the bottom of the screen capture. Add a comfortable margin to this number (for example, 20%), and use that time in seconds as the sample interval for that object. If multiple instances of HP Command View EVAPerf will be executed, use the longest interval for each of the sample sets. It is always preferable to have the sample rate determined by timed execution, as opposed to continuously running, which would happen if the sample interval is set too low.

Organizing HP Command View EVAPerf output data for analysis

HP Command View EVAPerf data can be viewed using the Microsoft® Windows® Performance Monitor (Perfmon) utility or through the HP Command View EVAPerf command line interface (CLI). In the current release of the product, most, but not all, metrics and ancillary information are available through Perfmon. Refer to the HP EVA documentation for a complete list of objects and counters supported in the Perfmon implementation. This paper discusses only the HP Command View EVAPerf CLI. All information regarding corresponding Perfmon performance objects and counters is still valid as are their interpretation and use.

You can select multiple output formats for the HP Command View EVAPerf data—text, CSV, and tab separated variable (TSV) formats. Selecting a continuous output mode (`-cont`) creates a text stream of performance data that can be viewed directly or redirected to an output file. This data is collected from all EVAs attached to the HP Command View EVAPerf host and for as many components as you want to monitor.

Performance objects and counters

The following sections provide examples of output data that have been captured in a plain text format for easy readability. However, if the data is to be further processed, HP recommends the CSV format. All data examples provided are generated using synthetic workloads and therefore show great variation between virtual disks but very little variability over time. These examples do not represent any real workload observed in a customer environment.

IMPORTANT: In all examples provided, there can be no inference that the numbers represent good, bad, or even valid performance information. All data is shown strictly as examples of output format.

The output data file consists of performance counters and their associated reference information. When viewing the output data file, the performance data is tabulated into a matrix of rows and columns. Some non-performance information is common to several objects. Those include:

- The Ctlr column indicates for which controller the metrics are being reported. The A or B nomenclature may or may not correlate to that displayed in Command View, so unambiguous identification must be done using controller serial numbers.
- The Node column indicates the EVA storage array from which specific data has been collected. In the examples in this paper, the node data has been abbreviated for clarity.
- The GroupID column indicates in which disk group that a virtual disk or physical disk is a member.

By default, HP Command View EVAPerf uses MB/s for data rates, which is equivalent to 1,000,000 bytes per second. There is an option for data rates to be presented in KB/s, which is equivalent to 1,024 bytes per second. HP Command View EVAPerf displays individual counters with the appropriate data rate metrics in the header. The default unit for latencies is milliseconds (ms), but latencies can also be specified in microseconds.

In some cases, the node or object name has been abbreviated for clarity. For example, the array node name `5000-1FE1-5001-8680` has been abbreviated to …`-8680.` The virtual disk WorldWide Name (WWN) `6005-08B4-0010-2E24-0000-7000-015E-0000` has been abbreviated to …`-015E`. Friendly names are much easier to work with but were not used when these examples were captured.

Thirteen performance objects are available in the HP Command View EVAPerf monitor: host ports, virtual disks, virtual disks by disk group, physical disks, physical disks by disk group, data replication tunnels, read latency histogram, write latency histogram, transfer size histogram, host connections, port status, array status, and controller status. Each of these performance objects includes a set of counters that characterize the workload and performance metrics for that object. These objects and counters and their meanings are defined in the following sections.

Comparing HP StorageWorks Enterprise Virtual Array 3000/5000 and HP StorageWorks Enterprise Virtual Array 4000/6000/8000 counters

The counters are managed somewhat differently in HP StorageWorks Enterprise Virtual Array 3000 (EVA3000) and HP StorageWorks Enterprise Virtual Array 5000 (EVA5000) systems than in EVA4000, EVA6000, and EVA8000 systems. In EVA3000 and EVA5000 systems, most counters (Req/s, MB/s, and so on) are one-second averages per sample, regardless of the interval between samples. Therefore, each sample represents an instantaneous snapshot of the activity at that moment. In EVA4000, EVA6000, and EVA8000 systems, however, the counters are true averages over the sample interval, so highly variable data will have different characteristics for longer samples than for shorter.

In EVA3000 and EVA5000 systems, count metrics such as HP Continuous Access tunnel retries and host connection busies are non-resetting incremental counters, while in EVA4000, EVA6000, and EVA8000 systems, the counters present the rate of counts that have been logged during the sample interval.

Host port statistics performance object

The EVA host port statistics collect information for each of the EVA host ports. These metrics provide information on the performance and data flow through each of these ports. The EVA8000 has four host ports per controller, eight ports per controller pair. All other controller models have two ports per controller, four ports per controller pair.

Enumerating requests per second, data rates and latencies for both reads and writes provide an immediate picture of the activity for each port and each controller. Queue depths by themselves should not be construed as good or bad, but rather an additional metric to help understand performance issues.

NOTE: All host port statistics include only host initiated activity. No remote mirror traffic is included in any of these metrics.

Host port statistics with related counters

```
Name  Read   Read    Read Write  Write   Write   Av. Ctlr Node
     Req/s   MB/s Latency Req/s   MB/s Latency Queue
                     (ms)                 (ms) Depth
---- ----- ------ ------- ----- ------ ------- ----- ---- -------------------
FP1    582   4.77    17.5   393   3.23    11.1    67 A    5000-1FE1-5001-8680
FP2      0   0.00     0.0   758  49.70     7.7    27 A    5000-1FE1-5001-8680
FP3    155   2.55    18.5     0   0.00     0.0    15 A    5000-1FE1-5001-8680
FP4   2587 169.55     2.1     0   0.00     0.0   246 A    5000-1FE1-5001-8680
FP1      0   0.00     0.0   683  44.77     8.6    29 B    5000-1FE1-5001-8680
FP2    584   4.79    16.5   400   3.28    11.8    13 B    5000-1FE1-5001-8680
FP3    155   2.55    16.8     0   0.00     0.0    90 B    5000-1FE1-5001-8680
FP4   2070 135.68     2.7     0   0.00     0.0     8 B    5000-1FE1-5001-8680
```

Read Req/s

This counter shows the read request rate from host-initiated requests that were sent through each host port for read commands.

For this document, the term "request" means an action initiated by a client to transfer a defined length of data to or from a specific storage location. With respect to the controller, the client is a host; with respect to a physical disk, the client is a controller.

Read MB/s

This counter tabulates the read data rate of host initiated requests that were sent through each host port for read commands.

All data rates in these examples are expressed in the default unit of MB/s. All rates can optionally be listed in KB/s.

Read Latency

Read latency is the elapsed time from when the EVA receives a read request until it returns completion of that request to the host client over the EVA host port. This time is an average of the read request latency for all virtual disks accessed through this port and includes both cache hits and misses.

Write Req/s

This counter tabulates the write request rate of host initiated requests that were sent through each host port for write commands.

Write MB/s

This counter tabulates the write data rate of host-initiated requests that were sent through each host port for write commands.

Write Latency

Write latency is the elapsed time from when the EVA receives a write request until it returns completion of that request to the host client over a specific EVA host port. This time is an average of the write request latency for all virtual disks accessed through this port.

Av Queue Depth

This counter tracks the average number of outstanding host requests against all virtual disks accessed through this EVA host port and over the sample interval. This metric includes all host-initiated commands, including non-data transfer commands.

EVA Virtual Disk performance object

The Virtual Disk object reports workload and performance for each virtual disk on the EVA. A virtual disk can also be a Snapshot, Snapclone, or replication volume. In the tool headers, the term "Logical Unit Number" (LUN) is used interchangeably with virtual disk.

Virtual disks must be presented to a host to be seen by HP Command View EVAPerf. However, remote mirror volumes on the replication system are visible without being presented.

Virtual disk statistics provide a wealth of information about the activity against each of the virtual disks in the storage system. Activity for each virtual disk is reported separately for each controller accessing that virtual disk. The total activity for each virtual disk is then the sum of the reported activity for each controller.

The EVA3000 and EVA5000 controllers function in Active-Standby mode, where the disk is accessed solely through the controller to which it is online. Therefore, all activity to a given virtual disk is through a single controller. The EVA4000, EVA6000, and EVA8000 controllers, however, support a feature called Active-Active mode in which one controller is the online, or preferred, controller for a given virtual disk (the owning controller) but requests can still be processed by the other controller (the proxy). All requests to the owning controller are processed in the same fashion as for an EVA3000 or EVA5000 controller. Read commands to the proxy controller are passed to the owning controller, which, in turn, retrieves the data from disk or cache. The data is then passed to the proxy controller through the mirror port, which then satisfies the host read request. Writes to the proxy controller are stored in local cache but copied across the mirror port, in the same manner as with EVA3000 and EVA5000 systems. The owning controller is responsible for cache flushes to physical disk. Write data is always mirrored in EVA4000, EVA6000, and EVA8000 systems. Therefore, there is little additional impact to write performance for a proxy controller in Active-Active mode, compared with writes to the owning controller.

In EVA4000, EVA6000, and EVA8000 systems, all host requests are logged only by the receiving controller, whether owning or proxy. Thus, all request rate and data rate activity for a given virtual disk is simply the respective sums for both controllers.

Information unique to virtual disk data

The "Online To" column defines which controller is the owning controller for each virtual disk.

"Wr Mode" indicates whether the specific controller is in cache write-back or write-through mode. While write-back is the default mode, you can specify write-through mode on an individual virtual disk basis. You must change write-back mode to write-through on each virtual disk before initiating a three-phase Snapshot or Snapclone operation against that virtual disk. For these cases, the mode reverts to write-back when the action is complete. The write mode also immediately changes to write-through if a controller detects a battery condition in which there is a possibility of not being able to guarantee cache integrity. That mode persists until that battery condition is corrected.

"Mirr" indicates whether cache mirroring is enabled for each virtual disk. For EVA4000, EVA6000, and EVA8000 controllers, cache mirroring is always enabled to support the Active-Active feature.

EVA virtual disk with related counters

```
ID Read   Read Read    Read  Read Read    Write Write Write   Flush Mirror Prefetch Group Online Mirr Wr   Ctlr LUN      Node
   Hit    Hit  Hit     Miss  Miss Miss    Req/s MB/s  Latency MB/s  MB/s   MB/s      ID    To          Mode
   Req/s  MB/s Latency Req/s MB/s Latency             (ms)
                (ms)               (ms)
-- -----  ------ ------ ------- ----- ------ ------- ----- ------ -------- ------ ------ -------- ----- ------ ---- ------ ----------------
0  0      0.00 0.0     143   1.17 27.0    97    0.80 19.5    0.00  0.00   0.00     Default B      Yes  Back A     …-015E …-8680
0  0      0.00 0.0       0   0.00  0.0     0    0.00  0.0    0.80  1.97   0.00     Default B      Yes  Back B     …-015E …-8680
1  0      0.00 0.0       0   0.00  0.0     0    0.00  0.0    0.00  0.00   0.00     Default B      Yes  Back A     …-0163 …-8680
1  0      0.00 0.0      39   0.64 15.5     0    0.00  0.0    0.00  0.00   0.00     Default B      Yes  Back B     …-0163 …-8680
2  0      0.00 0.0     117   7.69 12.3     0    0.00  0.0    0.00  0.00   0.00     Default B      Yes  Back A     …-0168 …-8680
2  0      0.00 0.0       0   0.00  0.0     0    0.00  0.0    0.00  7.69   7.70     Default B      Yes  Back B     …-0168 …-8680
```

Read Hit Req/s

This counter reports the rate at which read requests are satisfied from the EVA cache memory. Data will be in the EVA read cache because of a previous cache miss or a prefetch operation generated by a sequential read data stream.

Read Hit MB/s

This counter reports the rate at which data is read from the EVA cache memory because of host virtual disk read hit requests.

Read Hit Latency

This counter reports the average time taken from when a read request is received until that request has been satisfied from the EVA cache memory.

Read Miss Req/s

This counter reports the rate at which host read requests could not be satisfied from the EVA cache memory and had to be read from physical disks.

Read Miss Data Rate

This counter reports the rate of read data that was not present in the EVA cache memory and had to be read from physical disks.

Read Miss Latency

This counter reports the average time from when a host read request is received until that request has been satisfied from the physical disks.

Write Req/s

This counter reports the rate of write requests to a virtual disk that were received from all hosts, transfers from an HP Continuous Access EVA source system to this system for data replication, and host data written to Snapshot or Snapclone volumes.

Write Data Rate

This counter reports the rate at which data is written to the virtual disk by all hosts and includes transfers from an HP Continuous Access EVA source system to this system for data replication.

Write Latency

This counter reports the time between when a write request is received from a host and when request completion is returned.

Flush Data Rate

This counter represents the rate at which host write data is written to physical storage on behalf of the associated virtual disk. The aggregate of flush counters for all virtual disks on both controllers is the rate at which data is written to the physical drives and is equal to the aggregate host write data.

Data replication data written to the destination volume is included in the flush statistics. Host writes to Snapshot and Snapclone volumes are included in the flush statistics, but data flow for internal Snapshot and Snapclone normalization and copy-before-write activity is not included.

Mirror Data Rate

NOTE: The mirror data rates referenced in this section refer only to the data across the mirror ports and are not related to the physical disk mirroring for Vraid1 redundancy.

This counter represents the rate at which data moves across the mirror port in servicing read and write requests to a virtual disk. Write data is always copied through the mirror port when cache mirroring is enabled for redundancy. The mirror data rate includes read data from the owning controller that must be returned to the requesting host through the proxy controller. Reported mirror traffic is always outbound from the referenced controller to the other controller.

If a write stream has been determined to be sequential access or if the virtual disk is Vraid 1 or 0, the mirror traffic for that stream includes only its write data. However, a random workload to a Vraid 5 volume requires parity to be copied as well, so the mirror data rate for that specific stream is double the host write data rate (one parity byte for every host data byte).

Host writes to Snapshot and Snapclone volumes are mirrored and included in the mirror data rate statistics for the replication volumes. Snapclone normalization writes and copy-before-write activity for both Snapshot and Snapclones are done by the owning controller and do not pass through the mirror port. Mirror rates for the virtual disk on the destination system for a remote mirror pair includes initial copy and merge traffic to the replication volumes on the destination system as per normal host write operations.

Prefetch Data Rate

This counter on the controller that owns the virtual disk reports the rate at which data is read from physical storage into cache in anticipation of subsequent reads when a sequential read stream is detected

HP Continuous Access EVA initial copy data traffic for a replicated virtual disk is reported by the owning controller in the source volume prefetch statistics.

Physical Disk performance object

The physical disk counters report information on each physical disk on the system that is or has been a member of a disk group. These counters record all the activity to each disk and include all disk traffic for a host data transfer, as well as internal system support activity. This activity includes metadata updates, cache flushes, prefetch, sparing, leveling, Snapclone and Snapshot support, and redundancy traffic such as parity reads and writes or mirror copy writes. Each controller's metrics are reported separately, so the total activity to each disk is the sum of both controllers' activity.

The ID of each disk is from an internal representation and is not definitive for disk identification.

Physical disk with related counters

```
ID  Drive Drive   Read  Read   Read    Write Write  Write   Enc Bay Grp     Ctlr    Node
    Queue Latency Req/s MB/s   Latency Req/s MB/s   Latency         ID
    Depth (ms)                 (ms)                 (ms)
--- ----- ------- ----- ------ ------- ----- ------ ------- --- --- --- ---- -----------
0    37     -     12    1.10    16.4     7   0.52   20.2      2   1 Default  A    …-8680
0    29     -     11    0.90    17.1   13    0.80   18.5      2   1 Default  B    …-8680
1    35     -     11    0.94    17.8     7   0.52   21.2      3   1 Default  A    …-8680
1    28     -     11    0.89    17.1   13    0.79   19.2      3   1 Default  B    …-8680
2    32     -     11    1.09    17.0     8   0.52   21.7      1   1 Default  A    …-8680
2    32     -     12    0.91    19.3   12    0.79   20.2      1   1 Default  B    …-8680
3    31     -     10    0.93    15.2     8   0.52   22.1      5   1 Default  A    …-8680
3    31     -     13    0.89    16.1   12    0.79   20.1      5   1 Default  B    …-8680
4    38     -     12    1.10    16.5     8   0.53   21.4      6   1 Default  A    …-8680
4    30     -     12    0.91    17.4   13    0.80   20.2      6   1 Default  B    …-8680
5    34     -     10    0.93    17.2     8   0.53   21.9      4   1 Default  A    …-8680
5    27     -     11    0.90    14.4   13    0.80   21.1      4   1 Default  B    …-8680
6    35     -     11    1.09    15.0     7   0.52   20.4     14   1 Default  A    …-8680
6    34     -     12    0.91    17.3   13    0.79   20.2     14   1 Default  B    …-8680
```

Drive Queue Depth

The queue depth metric is the average number of requests, both read and write, in process from each controller.

Drive Latency

This counter, which is not in EVA4000, EVA6000, or EVA8000 systems, reports the average time between when a data transfer command is sent to a disk and when command completion is returned from the disk. This time is not broken into read and write latencies but is a "command processing" time. Completion of a disk command does not necessarily imply host request completion because the request to a specific physical disk might be only a part of a larger request operation to a virtual disk.

Read Req/s

This counter reports the rate at which read requests are sent to the disk drive for each controller.

Read MB/s

This counter reports the rate at which data is read from a drive for each controller.

Read Latency

This counter, which is not in EVA3000 and EVA5000 systems, reports the average time for a physical disk to complete a read request from each controller.

Write Req/s

This counter reports the rate at which write requests are sent to the disk drive from each controller.

Write MB/s

This counter reports the rate at which data is sent to a drive from each controller.

Write Latency

This counter, which is not in EVA3000 and EVA5000 systems, reports the average time for a physical disk to complete a write request from each controller.

Enc

This is the number for the drive enclosure that holds the referenced drive. The numbering refers to that reported by the EVA controller and is reported by HP StorageWorks Command View EVA.

Bay

This is enclosure slot number in which the drive resides.

HP Continuous Access Tunnel performance object

The EVA HP Continuous Access tunnel counters report the intensity and behavior of the link traffic between the source and destination EVAs. The HP Continuous Access tunnel statistics have values only if there is at least one active HP Continuous Access EVA group on the storage array. Otherwise, only a header prints.

The first five columns define the physical and logical representation for each tunnel. Two systems are represented in this data capture, the source and destination system on each end of the tunnels. In this example, the active tunnel is the one tunnel that shows activity, 16 MB/s of write activity in this case. It is tunnel 3 on host port 3 of controller B on the source side (node -8680) and tunnel 2 on host port 2 of controller B on the destination side (node -2350). The hex source and destination IDs are with respect to each controller and should correlate between controllers.

On EVA4000, EVA6000, and EVA8000 controllers, there can be up to four tunnels open on a given host port. More than one tunnel can be reported as open, but for a single HP Continuous Access EVA group, only one will be active. Multiple HP Continuous Access EVA groups can share the same tunnel. Statistics for each tunnel are reported by both the source EVA and the destination, except that the directional counters will be complementary. For example, `Write Out MB/s` for one controller will be duplicated as `Write In MB/s` on the other.

HP Continuous Access tunnel statistics with related counters

```
Ctlr Tunnel Host Source  Dest.  Round  Copy    Write   Copy   Copy   Write  Write   Node
     Number Port ID      ID     Trip   Retries Retries In     Out    In     Out
                                Delay                  MB/s   MB/s   MB/s   MB/s
                                (ms)
---- ------ ---- ------ ------ ------  ------- ------- ------ ------ ------ ------ -------
 A       2     2  040900 030700    4.4       0       0  0.00   0.00   0.00   0.00 ..-8680
 A       3     2  040900 030900   25.5       0       0  0.00   0.00   0.00   0.00 ..-8680
 B       3     3  040800 030700   17.5       0       0  0.00   0.00   0.00  16.16 ..-8680
 A       3     3  030900 040900   10.9       0       0  0.00   0.00   0.00   0.00 ..-2350
 B       2     2  030700 040800   10.2       0       0  0.00   0.00  16.18   0.00 ..-2350
 B       3     2  030700 040900   11.4       0       0  0.00   0.00   0.00   0.00 ..-2350
```

Round Trip Delay

The Round Trip Delay counter shows the average time in milliseconds during the measurement interval for a signal (ping) to travel from source to destination and back again. It includes the intrinsic latency of the link round trip time and any queuing encountered from the workload. Thus, the Round Trip Time counter measures the link delay if there is no data flow between source and destination. If there is replication traffic on the link, then the ping can be queued behind any data transmissions already in process, thus increasing round trip delay. If the destination controller is busy, then the round trip time is extended accordingly. Round trip delay is reported for each active tunnel for both local and remote systems. A value of 10,000 is reported when the tunnel is initially opened and then is updated if successful pings are executed. A value of 10,000 is also shown for any tunnel interrupted by a link problem. Round trip delay is reported for all open tunnels, whether active or inactive.

Copy Retries

Copy Retries is a count of all copy actions from the source EVA that had to be retransmitted during the sample interval in response to a failed copy transaction. Each retry results in an additional copy of 128 KB. Retries are reported by both the local and remote systems.

Write Retries

Write Retries is a count of all write actions from the source EVA that had to be retransmitted during the sample interval in response to a failed write to the replication volume. Each retry results in an additional copy of 8 KB. If the replication write consists of multiple 8 KB segments, only the failed segments are retransmitted. Retries are reported by both the local and remote systems.

Copy In MB/s

The Copy In MB/s counter measures the inbound data from the source EVA to create a remote mirror on the destination EVA during initial copy or from a full copy initiated by the user or from a link event. Copy retries are included.

Copy Out MB/s

The Copy Out MB/s counter is identical to the corresponding Copy In MB/s counter, except that the direction is outbound from the local source EVA to the remote destination EVA.

Write In MB/s

The Write In MB/s counter measures the inbound data rate from the source EVA to the local destination EVA for host writes, merges, and replication write retries. A merge is an action initiated by the source controller to write new host data that has been received and logged while replication writes to the destination EVA were interrupted and now have been restored.

Write Out MB/s

The Write Out MB/s counter is identical to the corresponding Write In MB/s counter, except that the direction is outbound from the local source EVA to the remote destination EVA.

EVA Host Connection performance object

The two counters in this object deal with the host port connections on the EVA and provide information for each external host adapter that is connected to the EVA.

The host connection metrics provide some information on the activity from each adapter seen as a host to the array. The average number of requests in process (queue depth) over the sample interval is reported, along with the number of busies that were issued to each adapter in the interval.

Host connection with related counters

```
Host      Port Queue Busies Ctlr Node
Name           Depth
--------- ---- ----- ------ ---- -------------------
C92B-65B1    -    14      0 A    5000-1FE1-5001-8680
C92B-67B9    -    41      0 A    5000-1FE1-5001-8680
C92B-68A3    -     1      0 A    5000-1FE1-5001-8680
C92B-679C    -     8      0 A    5000-1FE1-5001-8680
C92B-67CE    -     0      0 B    5000-1FE1-5001-8680
C92B-638B    -     7      0 B    5000-1FE1-5001-8680
C92B-6836    -     3      0 B    5000-1FE1-5001-8680
C92B-671F    -    30      0 B    5000-1FE1-5001-8680
```

Port

This counter, which is not in EVA4000, EVA6000, and EVA8000 systems, is the host port number on the controller through which the host connection is made.

Queue Depth

This counter records the average number of outstanding requests from each of the corresponding host adapters.

Busies

This counter represents the number of busy responses sent to a specific host during the sample interval. Busies represent a request from the controller to the host to cease I/O traffic until some internal job queue is reduced.

Histogram metrics performance objects

Each of the preceding metrics reports a single average value for each counter over the sample interval. Histograms provide a broader description of those metrics by indicating the range of values observed in each interval and the frequency with which different values were observed. In a complex system such as the EVA, some variation in a given metric is normal. Histograms enable you to assess that variability as normal or problematic. Histogram metrics are reported for each virtual disk for each controller in the EVA4000, EVA6000, and EVA8000. Histograms are not available in the EVA3000 and EVA5000.

Read or write latency histogram

The latency histograms provide insight into the distribution of response times for requests. The read latency distribution combines both read hits and read misses and sometimes shows the bimodal distribution associated with the two processes. Distributions also show how well the response times are grouped or if they have values across many levels of latency. The histograms consist of a set of 10 ranges of milliseconds in which the counter reports the number of requests that exhibited latency in the given range during the sample interval. For example, in the following histogram, there were 1,229 requests to virtual disk 0, and which experienced a latency of at least 13 ms but less than 26 ms in the sample interval.

Distributions are reported individually for each virtual disk by each controller providing the host service. It is therefore necessary to sum the statistics from each controller to accurately report the full histogram counts.

All histogram buckets are fixed according to an exponentially increasing scale, so finer granularity is provided at the lower end of the scale, and coarser granularity is provided at the upper end. Latency units are milliseconds, and each of the 10 ranges is fixed.

The following example was taken from read latency histograms, but it is indistinguishable from the write latency in format and information. Therefore, only one example of latency histogram is presented.

Read or write latencies histogram

```
ID     0   1.6   3.3   6.6    13    26    52   105   210   420 Ctlr     LUN      Node
       -     -     -     -     -     -     -     -     -     -
     1.6   3.3   6.6    13    26    52   105   210   420   INF
-- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ---- -------------------
 0     0     0     0    15  1229   889    52     6     0     0 A    ...-015E ...-8680
 0     0     0     0     0     0     0     0     0     0     0 B    ...-015E ...-8680
 1     0     0     0     0     0     0     0     0     0     0 A    ...-0163 ...-8680
 1     6    57    23   146   248    67    12     1     0     0 B    ...-0163 ...-8680
 2     0     0    16  1152   636     0     0     0     0     0 A    ...-0168 ...-8680
 2     0     0     0     0     0     0     0     0     0     0 B    ...-0168 ...-8680
 3     0     0     0     0     0     0     0     0     0     0 A    ...-016D ...-8680
 3     0     0     0     0     0     0     0     0     0     0 B    ...-016D ...-8680
 4     0     0     0     0     0     0     0     0     0     0 A    ...-0172 ...-8680
 4     0     0     0     0     0     0     0     0     0     0 B    ...-0172 ...-8680
 5     0     0     0     0     0     0     0     0     0     0 A    ...-0177 ...-8680
 5    11   147   157   721   920   194    25     3     0     0 B    ...-0177 ...-8680
 6     0     0     0     1   329   221    16     2     0     0 A    ...-017C ...-8680
 6     0     0     0     0     0     0     0     0     0     0 B    ...-017C ...-8680
 7     0     0     0     0     0     0     0     0     0     0 A    ...-0181 ...-8680
 7   852  5304  1647    44     1     0     0     0     0     0 B    ...-0181 ...-8680
 8     0     2   133  1256   618   197    17     2     0     0 A    ...-0186 ...-8680
 8     0     0     0     0     0     0     0     0     0     0 B    ...-0186 ...-8680
```

Transfer size histogram

Transfer size distributions are somewhat different than that for latencies because transfer size distributions are defined by the application workload and are, in essence, deterministic values. Latencies show how the system responds to a given workload, but transfer sizes show the makeup of that workload.

In the following example, the synthetic workload used to capture this data used a single transfer size for each virtual disk's load, and that is clearly shown in each line. Real workloads usually show more variation than this example. Most operating systems have an upper limit of transfer size. Larger transfers can be specified at the application level, but the operating system breaks those transfers into multiple smaller chunks. For example, if the maximum transfer size is 128 KB and the application tries to write a 512-KB chunk, the controller will see four transfers of the smaller size. For this case, array statistics will not match host statistics for the same operation.

Transfer size histogram

```
ID     0    2K    4K    8K   16K   32K   64K  128K  256K  512K Ctlr   LUN     Node
       -     -     -     -     -     -     -     -     -     -
      2K    4K    8K   16K   32K   64K  128K  256K  512K   INF
-- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ---- ---------------
 0     0     0     0  3512     0     0     0     0     0     0 A    -015E ...-8680
 0     0     0     0     0     0     0     0     0     0     0 B    -015E ...-8680
 1     0     0     0     0     0     0     0     0     0     0 A    -0163 ...-8680
 1     0     0     0     0   582     0     0     0     0     0 B    -0163 ...-8680
 2     0     0     0     0     0     0  1753     0     0     0 A    -0168 ...-8680
 2     0     0     0     0     0     0     0     0     0     0 B    -0168 ...-8680
 3     0     0     0     0     0     0     0     0     0     0 A    -016D ...-8680
 3     0     0     0     0     0     0  1819     0     0     0 B    -016D ...-8680
 4     0     0     0     0     0     0  1440     0     0     0 A    -0172 ...-8680
 4     0     0     0     0     0     0     0     0     0     0 B    -0172 ...-8680
 5     0     0     0     0     0     0     0     0     0     0 A    -0177 ...-8680
 5     0     0     0  3600     0     0     0     0     0     0 B    -0177 ...-8680
 6     0     0     0     0   567     0     0     0     0     0 A    -017C ...-8680
 6     0     0     0     0     0     0     0     0     0     0 B    -017C ...-8680
 7     0     0     0     0     0     0     0     0     0     0 A    -0181 ...-8680
 7     0     0     0     0     0     0  7706     0     0     0 B    -0181 ...-8680
 8     0     0     0  3631     0     0     0     0     0     0 A    -0186 ...-8680
 8     0     0     0     0     0     0     0     0     0     0 B    -0186 ...-8680
```

Array Status performance object

The Array Status object reports basic workload to the overall storage system.

Array status with related counters

```
Total  Total Node
Host   Host
Req/s   MB/s
----- ------ -------------------
  894   1.01 5000-1FE1-5001-8680
    0   0.00 5000-1FE1-0015-2350
```

The array status counters provide an immediate representation of the total workload to a controller pair in request rate and data rate.

Controller Status performance object

The controller status output shows the controller processor utilizations and host data service utilizations. Controller status metrics for two storage systems are reported in the following example.

Controller status with related counters

```
CPU  Data  Ctlr   Serial          Node
 %    %
---  ----  ----   --------------  -------------------
 84    82    A    P8398FXAAR2003  5000-1FE1-5001-8680
 99    97    B    P8398FXAAR200D  5000-1FE1-5001-8680
  3     3    A    P8398FXAAR301Q  5000-1FE1-0015-2350
 12     8    B    P8398FXAAQZ001  5000-1FE1-0015-2350
```

CPU %

This counter presents the percentage of time that the CPU on the EVA controller is not idle. A completely idle controller shows 0%, while one that is saturated shows a value of 100%.

Data %

Percent data transfer time is the same as CPU %, except that it does not include time for any internal process that is not associated with a host initiated data transfer. Thus, it does not include background tasks such as sparing, leveling, Snapclone, Snapshot, remote mirror traffic, virtual disk management, or communication with tools such as HP Command View EVAPerf.

Port Status performance object

The port status counters report the health and error condition status for all fibre channel ports on each controller and are not performance related.

Port status with related counters

```
Name  Type   Status Loss   Bad  Loss  Link Rx   Discard Bad Proto Ctlr Node
                    of     Rx   of    Fail EOFa Frames  CRC Err
                    Signal Char Synch
----- ------ ------ ------ ---- ----- ---- ---- ------- --- ----- ---- ------------
DP-1A Device Up        0    0     0    0    0       0   0     0  A    5000-…-8680
DP-1B Device Up        0    0     0    0    0       0   0     0  A    5000-…-8680
DP-2A Device Up        0    0     0    0    0       0   0     0  A    5000-…-8680
DP-2B Device Up        0    0     0    0    0       0   0     0  A    5000-…-8680
MP1   Mirror Up        0    0     0    0    0       0   0     0  A    5000-…-8680
MP2   Mirror Up        0    0     0    0    0       0   0     0  A    5000-…-8680
FP1   Host   Up        0    0     0    0    0       0   0     0  A    5000-…-8680
FP2   Host   Up        0    0     0    0    0       0   0     0  A    5000-…-8680
FP3   Host   Up        0    0     0    0    0       0   0     0  A    5000-…-8680
FP4   Host   Up        0    0     0    0    0       0   0     0  A    5000-…-8680
DP-1A Device Up        0    0     0    0    0       0   0     0  B    5000-…-8680
DP-1B Device Up        0    0     0    0    0       0   0     0  B    5000-…-8680
DP-2A Device Up        0    0     0    0    0       0   0     0  B    5000-…-8680
DP-2B Device Up        0    0     0    0    0       0   0     0  B    5000-…-8680
MP1   Mirror Up        0    0     0    0    0       0   0     0  B    5000-…-8680
MP2   Mirror Up        0    0     0    0    0       0   0     0  B    5000-…-8680
FP1   Host   Up        0    0     0    0    0       0   0     0  B    5000-…-8680
FP2   Host   Up        0    0     0    0    0       0   0     0  B    5000-…-8680
FP3   Host   Up        0    0     0    0    0       0   0     0  B    5000-…-8680
FP4   Host   Up        0    0     0    0    0       0   0     0  B    5000-…-8680
```

Viewing individual performance objects

While the plain text output data provides a general sense of performance over a short period, additional output file processing is required to facilitate performance analysis over a longer period and when graphing a performance counter value over time. This additional processing is also required to derive performance of components that are not directly represented in HP Command View EVAPerf data such as controller level workload.

When gathering performance data for many components over an extended period, the text file containing the performance data from HP Command View EVAPerf can become very large and difficult to analyze.

The original raw output file can provide a more sophisticated and in-depth performance analysis by organizing the performance object data into separate files for each performance object. Text processing tools, such as Perl and Awk, can help you with this task, which enables you to view a time series representation of each reported component for analysis and graphing.

Separating EVA storage subsystems

If you have multiple EVA storage subsystems (controller pairs) represented in one output file, separate the data in the output file into multiple files, each containing data for a single EVA storage subsystem. When you have each EVA storage subsystem represented in its own file, use the following procedures to organize your performance data for component-level analysis for each storage system.

NOTE: The –sz command line switch restricts data capture to only the systems specified. Multiple concurrent executions of HP Command View EVAPerf can be performed to capture each system in a separate file, but care in setup and execution is required if the object samples in one file must be time-correlated to those in another file.

Host port statistics

To effectively view host port statistics, extract all host port data and present it chronologically for each port.

1. Extract the port statistics from the HP Command View EVAPerf output file into one file of all host port data.

Example of host port statistics extracted from the output data file for both controllers

```
Time,Name,Read Req/s,Read MB/s,Read Latency (ms),Write Req/s,Write MB/s,Write Latency
(ms),Av. Queue Depth,Ctlr,Node

14/Mar/2005 8:43:42,FP1,579,4.75,17.6,394,3.23,11.3,5073,A,5000-1FE1-5001-8680
14/Mar/2005 8:43:42,FP2,0,0.00,0.0,754,49.46,7.7,1904,A,5000-1FE1-5001-8680
14/Mar/2005 8:43:42,FP3,155,2.55,18.4,0,0.00,0.0,923,A,5000-1FE1-5001-8680
14/Mar/2005 8:43:42,FP4,2570,168.44,2.1,0,0.00,0.0,1677,A,5000-1FE1-5001-8680
14/Mar/2005 8:43:42,FP1,0,0.00,0.0,680,44.59,8.6,1865,B,5000-1FE1-5001-8680
14/Mar/2005 8:43:42,FP2,580,4.76,16.5,397,3.26,11.8,4754,B,5000-1FE1-5001-8680
14/Mar/2005 8:43:42,FP3,157,2.58,17.0,0,0.00,0.0,894,B,5000-1FE1-5001-8680
14/Mar/2005 8:43:42,FP4,2074,135.93,2.7,0,0.00,0.0,1724,B,5000-1FE1-5001-8680
14/Mar/2005 8:43:52,FP1,594,4.87,17.8,415,3.41,11.5,160,A,5000-1FE1-5001-8680
14/Mar/2005 8:43:52,FP2,0,0.00,0.0,768,50.37,7.8,40,A,5000-1FE1-5001-8680
14/Mar/2005 8:43:52,FP3,159,2.61,18.5,0,0.00,0.0,21,A,5000-1FE1-5001-8680
14/Mar/2005 8:43:52,FP4,2631,172.47,2.1,0,0.00,0.0,37,A,5000-1FE1-5001-8680
14/Mar/2005 8:43:52,FP1,0,0.00,0.0,625,41.02,8.6,36,B,5000-1FE1-5001-8680
14/Mar/2005 8:43:52,FP2,546,4.47,17.0,372,3.05,11.7,146,B,5000-1FE1-5001-8680
14/Mar/2005 8:43:52,FP3,144,2.37,16.7,0,0.00,0.0,10,B,5000-1FE1-5001-8680
14/Mar/2005 8:43:52,FP4,1906,124.93,2.7,0,0.00,0.0,32,B,5000-1FE1-5001-8680
14/Mar/2005 8:44:2,FP1,569,4.66,17.7,403,3.30,11.5,105,A,5000-1FE1-5001-8680
14/Mar/2005 8:44:2,FP2,0,0.00,0.0,752,49.32,7.7,40,A,5000-1FE1-5001-8680
14/Mar/2005 8:44:2,FP3,156,2.57,18.5,0,0.00,0.0,27,A,5000-1FE1-5001-8680
14/Mar/2005 8:44:2,FP4,2572,168.61,2.1,0,0.00,0.0,36,A,5000-1FE1-5001-8680
14/Mar/2005 8:44:2,FP1,0,0.00,0.0,680,44.63,8.6,40,B,5000-1FE1-5001-8680
14/Mar/2005 8:44:2,FP2,580,4.75,16.6,396,3.25,11.9,183,B,5000-1FE1-5001-8680
14/Mar/2005 8:44:2,FP3,151,2.48,17.1,0,0.00,0.0,6,B,5000-1FE1-5001-8680
14/Mar/2005 8:44:2,FP4,2055,134.68,2.7,0,0.00,0.0,37,B,5000-1FE1-5001-8680
```

2. Rearrange the extracted data so that all of the ports for a single timestamp are grouped on a single line.

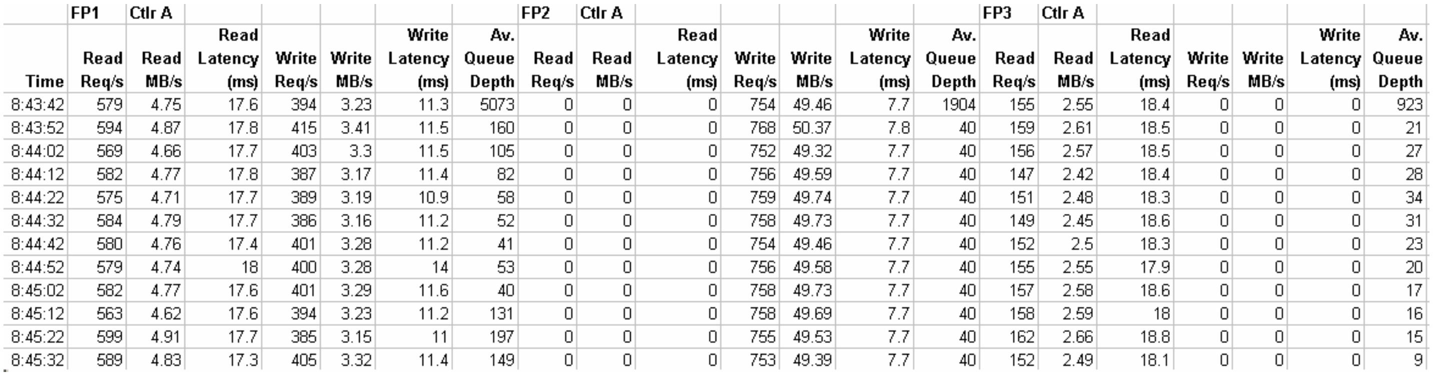
It might be helpful in a performance analysis effort to determine estimated transfer sizes at the host port level or the virtual disk level. This estimation is an average for the sample interval, calculated by dividing the data rate by the request rate for that interval. The unit for this calculation is MB/transfer. Multiplying by 1,000,000 and dividing by 1,024 yields units of KB/transfer.

Example of host port data grouped by timestamp (All eight host ports appear on one line with a single timestamp.)

```
,FP1,Ctlr A,,,,,,FP2,Ctlr A,,,,,,Fp3,Ctlr A,,,,,,FP4,Ctlr A,,,,,,FP1,Ctlr B,,,,,,FP2,Ctlr B,,,,,,FP3,Ctlr B,,,,,,FP4,Ctlr
B,,,,,
Time,Read Req/s,Read MB/s,Read Latency (ms),Write Req/s,Write MB/s,Write Latency (ms),Av. Queue Depth,Read Req/s,Read MB/s,Read
Latency (ms),Write Req/s,Write MB/s,Write Latency (ms),Av. Queue Depth,Read Req/s,Read MB/s,Read Latency (ms),Write Req/s,Write
MB/s,Write Latency (ms),Av. Queue Depth,Read Req/s,Read MB/s,Read Latency (ms),Write Req/s,Write MB/s,Write Latency (ms),Av.
Queue Depth,Read Req/s,Read MB/s,Read Latency (ms),Write Req/s,Write MB/s,Write Latency (ms),Av. Queue Depth,Read Req/s,Read
MB/s,Read Latency (ms),Write Req/s,Write MB/s,Write Latency (ms),Av. Queue Depth,Read Req/s,Read MB/s,Read Latency (ms),Write
Req/s,Write MB/s,Write Latency (ms),Av. Queue Depth,Read Req/s,Read MB/s,Read Latency (ms),Write Req/s,Write MB/s,Write Latency
(ms),Av. Queue Depth
8:43:42,579,4.75,17.6,394,3.23,11.3,5073,0,0,0,754,49.46,7.7,1904,155,2.55,18.4,0,0,0,923,2570,168.44,2.1,0,0,0,1677,0,0,0,680,
44.59,8.6,1865,580,4.76,16.5,397,3.26,11.8,4754,157,2.58,17,0,0,0,894,2074,135.93,2.7,0,0,0,1724
8:43:52,594,4.87,17.8,415,3.41,11.5,160,0,0,0,768,50.37,7.8,40,159,2.61,18.5,0,0,0,21,2631,172.47,2.1,0,0,0,37,0,0,0,625,41.02,
8.6,36,546,4.47,17,372,3.05,11.7,146,144,2.37,16.7,0,0,0,10,1906,124.93,2.7,0,0,0,32
8:44:02,569,4.66,17.7,403,3.3,11.5,105,0,0,0,752,49.32,7.7,40,156,2.57,18.5,0,0,0,27,2572,168.61,2.1,0,0,0,36,0,0,0,680,44.63,8
.6,40,580,4.75,16.6,396,3.25,11.9,183,151,2.48,17.1,0,0,0,6,2055,134.68,2.7,0,0,0,37
8:44:12,582,4.77,17.8,387,3.17,11.4,82,0,0,0,756,49.59,7.7,40,147,2.42,18.4,0,0,0,28,2574,168.73,2.1,0,0,0,36,0,0,0,684,44.89,8
.5,41,581,4.77,16.8,398,3.26,11.7,181,158,2.6,16.7,0,0,0,6,2060,135.04,2.7,0,0,0,36
8:44:22,575,4.71,17.7,389,3.19,10.9,58,0,0,0,759,49.74,7.7,40,151,2.48,18.3,0,0,0,34,2574,168.75,2.1,0,0,0,35,0,0,0,684,44.88,8
.5,39,589,4.83,16.7,388,3.18,11.5,99,152,2.5,16.6,0,0,0,37,2065,135.39,2.7,0,0,0,37
8:44:32,584,4.79,17.7,386,3.16,11.2,52,0,0,0,758,49.73,7.7,40,149,2.45,18.6,0,0,0,31,2575,168.8,2.1,0,0,0,34,0,0,0,688,45.1,8.5
,38,592,4.85,16.7,383,3.14,11.9,100,160,2.62,16.6,0,0,0,30,2064,135.28,2.7,0,0,0,39
8:44:42,580,4.76,17.4,401,3.28,11.2,41,0,0,0,754,49.46,7.7,40,152,2.5,18.3,0,0,0,23,2567,168.24,2.1,0,0,0,35,0,0,0,679,44.5,8.6
,39,573,4.7,16.5,404,3.31,11.7,39,160,2.63,16.9,0,0,0,11,2062,135.15,2.7,0,0,0,35
8:44:52,579,4.74,18,400,3.28,14,53,0,0,0,756,49.58,7.7,40,155,2.55,17.9,0,0,0,20,2567,168.27,2.1,0,0,0,35,0,0,0,683,44.82,8.5,4
1,586,4.81,16.8,394,3.23,14.8,38,156,2.57,16.4,0,0,0,34,2079,136.27,2.6,0,0,0,34
8:45:02,582,4.77,17.6,401,3.29,11.6,40,0,0,0,758,49.73,7.7,40,157,2.58,18.6,0,0,0,17,2568,168.34,2.1,0,0,0,34,0,0,0,686,45.02,8
.5,40,572,4.69,16.2,395,3.24,12,39,152,2.5,16.5,0,0,0,24,2067,135.51,2.7,0,0,0,38
8:45:12,563,4.62,17.6,394,3.23,11.2,131,0,0,0,758,49.69,7.7,40,158,2.59,18,0,0,0,16,2580,169.09,2.1,0,0,0,36,0,0,0,682,44.72,8.
5,40,575,4.72,16.5,396,3.25,11.8,29,159,2.61,16.3,0,0,0,20,2061,135.07,2.7,0,0,0,35
8:45:22,599,4.91,17.7,385,3.15,11,197,0,0,0,755,49.53,7.7,40,162,2.66,18.8,0,0,0,15,2567,168.23,2.1,0,0,0,36,0,0,0,681,44.66,8.
5,39,591,4.85,16.7,407,3.33,11.8,58,158,2.6,17,0,0,0,16,2060,135.04,2.7,0,0,0,37
8:45:32,589,4.83,17.3,405,3.32,11.4,149,0,0,0,753,49.39,7.7,40,152,2.49,18.1,0,0,0,9,2571,168.55,2.1,0,0,0,36,0,0,0,678,44.45,8
.6,38,586,4.8,16.5,395,3.24,11.7,154,156,2.56,16.6,0,0,0,8,2060,135.04,2.7,0,0,0,38
```

The preceding illegible example is included to present the extraction of host port data as time series information as a tool would process it. The following sample is exactly the same information in spreadsheet form to more clearly show content.

Example of host port data grouped by timestamp from previous example in a spreadsheet format (This spreadsheet illustrates only the first three host ports for controller A).



	FP1	Ctlr A						FP2	Ctlr A						FP3	Ctlr A					
Time	Read Req/s	Read MB/s	Read Latency (ms)	Write Req/s	Write MB/s	Write Latency (ms)	Av. Queue Depth	Read Req/s	Read MB/s	Read Latency (ms)	Write Req/s	Write MB/s	Write Latency (ms)	Av. Queue Depth	Read Req/s	Read MB/s	Read Latency (ms)	Write Req/s	Write MB/s	Write Latency (ms)	Av. Queue Depth
8:43:42	579	4.75	17.6	394	3.23	11.3	5073	0	0	0	754	49.46	7.7	1904	155	2.55	18.4	0	0	0	923
8:43:52	594	4.87	17.8	415	3.41	11.5	160	0	0	0	768	50.37	7.8	40	159	2.61	18.5	0	0	0	21
8:44:02	569	4.66	17.7	403	3.3	11.5	105	0	0	0	752	49.32	7.7	40	156	2.57	18.5	0	0	0	27
8:44:12	582	4.77	17.8	387	3.17	11.4	82	0	0	0	756	49.59	7.7	40	147	2.42	18.4	0	0	0	28
8:44:22	575	4.71	17.7	389	3.19	10.9	58	0	0	0	759	49.74	7.7	40	151	2.48	18.3	0	0	0	34
8:44:32	584	4.79	17.7	386	3.16	11.2	52	0	0	0	758	49.73	7.7	40	149	2.45	18.6	0	0	0	31
8:44:42	580	4.76	17.4	401	3.28	11.2	41	0	0	0	754	49.46	7.7	40	152	2.5	18.3	0	0	0	23
8:44:52	579	4.74	18	400	3.28	14	53	0	0	0	756	49.58	7.7	40	155	2.55	17.9	0	0	0	20
8:45:02	582	4.77	17.6	401	3.29	11.6	40	0	0	0	758	49.73	7.7	40	157	2.58	18.6	0	0	0	17
8:45:12	563	4.62	17.6	394	3.23	11.2	131	0	0	0	758	49.69	7.7	40	158	2.59	18	0	0	0	16
8:45:22	599	4.91	17.7	385	3.15	11	197	0	0	0	755	49.53	7.7	40	162	2.66	18.8	0	0	0	15
8:45:32	589	4.83	17.3	405	3.32	11.4	149	0	0	0	753	49.39	7.7	40	152	2.49	18.1	0	0	0	9

3. Graph the individual port data over the time interval of interest.

Virtual disk statistics

To view the virtual disk performance statistics, extract the data for each virtual disk into a separate time series file by performing the following suggested steps.

1. Combine read hit and read miss statistics into single set of metrics for all reads (optional).
2. Organize all virtual disk statistics for each controller into separate files or separate fields in one file.
3. Produce additional information by calculating transfer sizes for read hits, read misses, and write misses for each virtual disk as described previously for host ports (optional).

Example of virtual disk performance data extracted from the full output data file or from the single object data file

```
Time,ID,Read Hit Req/s,Read Hit MB/s,Read Hit Latency (ms),Read Miss Req/s,Read Miss
MB/s,Read Miss Latency (ms),Write Req/s,Write MB/s,Write Latency (ms),Flush RPS,Flush
MB/s,Mirror MB/s,Prefetch MB/s,Group ID,Online To,Mirr,Wr Mode,Ctlr,LUN

9:43:18,0,0,0,0,144,1.19,26.8,98,0.8,18.6,-,0,0,0,256,B,Yes,Back,A,…-015E
9:43:28,0,0,0,0,131,1.08,26.8,88,0.72,18.2,-,0,0,0,256,B,Yes,Back,A,…-015E
9:43:38,0,0,0,0,142,1.17,26.9,100,0.82,17.4,-,0,0,0,256,B,Yes,Back,A,…-015E
9:43:48,0,0,0,0,141,1.16,27.1,92,0.76,17.8,-,0,0,0,256,B,Yes,Back,A,…-015E
…

9:43:18,0,0,0,0,0,0,0,0,0,0,-,0.8,1.98,0,256,B,Yes,Back,B,…-015E
9:43:28,0,0,0,0,0,0,0,0,0,0,-,0.83,2.03,0,256,B,Yes,Back,B,…-015E
9:43:38,0,0,0,0,0,0,0,0,0,0,-,0.83,1.97,0,256,B,Yes,Back,B,…-015E
9:43:48,0,0,0,0,0,0,0,0,0,0,-,0.82,1.95,0,256,B,Yes,Back,B,…-015E
9:43:58,0,0,0,0,0,0,0,0,0,0,-,0.83,1.98,0,256,B,Yes,Back,B,…-015E
…

9:43:18,1,0,0,0,0,0,0,0,0,0,-,0,0,0,256,B,Yes,Back,A,…-0163
9:43:28,1,0,0,0,0,0,0,0,0,0,-,0,0,0,256,B,Yes,Back,A,…-0163
9:43:38,1,0,0,0,0,0,0,0,0,0,-,0,0,0,256,B,Yes,Back,A,…-0163
9:43:48,1,0,0,0,0,0,0,0,0,0,-,0,0,0,256,B,Yes,Back,A,…-0163
…

9:43:18,1,0,0,0,38,0.63,15.4,0,0,0,-,0,0,0,256,B,Yes,Back,B,…-0163
9:43:28,1,0,0,0,42,0.7,14.8,0,0,0,-,0,0,0,256,B,Yes,Back,B, ,…-0163
9:43:38,1,0,0,0,39,0.64,15.4,0,0,0,-,0,0,0,256,B,Yes,Back,B,…-0163
9:43:48,1,0,0,0,38,0.63,14.4,0,0,0,-,0,0,0,256,B,Yes,Back,B,…-0163
```

In the preceding example, the first four samples for virtual disk 0 and 1 are presented for each controller. Depending on what information you need, in most situations, it is advisable to combine the data for both controllers for each virtual disk. This procedure provides the complete activity picture for each virtual disk. However, in many circumstances, it is also helpful to combine all virtual disk data for each controller.

Read hits and read misses are reported separately for the virtual disks. For a simpler presentation of the virtual disk read activity, combine read hit and read miss statistics into a single set of metrics for all reads.

For further analysis, average all request rates and data rates for each virtual disk separately for the measurement interval for selection of those with dominant workloads. If latencies are to be merged, they must always be combined as weighted averages using following formula:

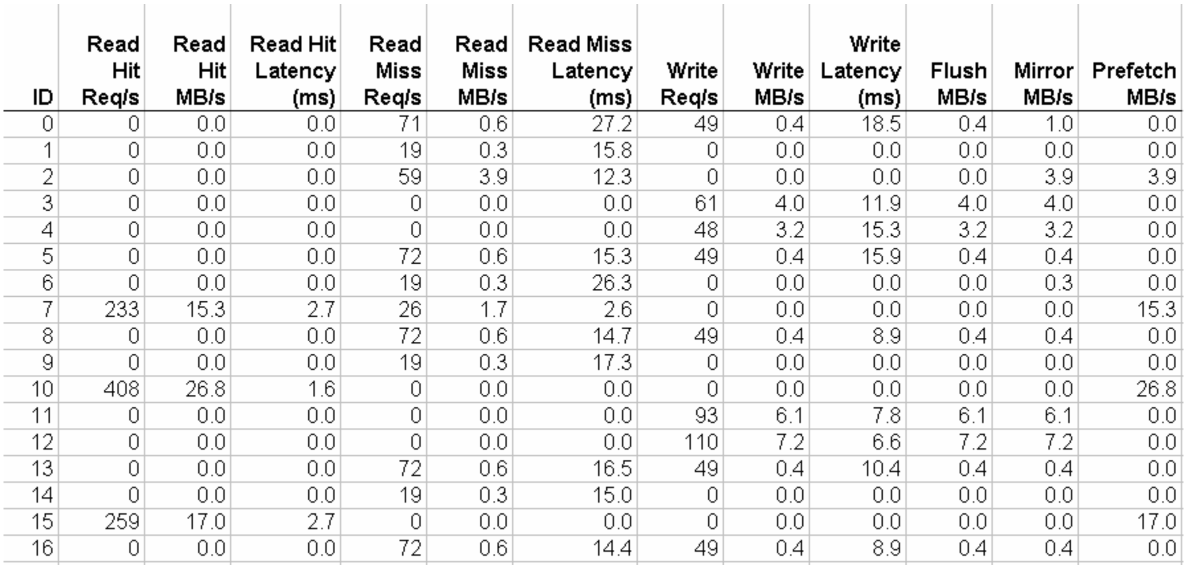
$$\text{Combined Latency} = \frac{(\text{Latency } 1 \times \text{Request Rate } 1) + (\text{Latency } 2 \times \text{Request Rate } 2)}{\text{Request Rate } 1 + \text{Request Rate } 2}$$

For example, if a given virtual disk reports read hit latency of 1.5 ms at a request rate of 80 requests/s and a read miss latency of 14.2 ms at a request rate of 327 requests/s, the combined read latency is $\frac{(1.5*80)+(14.2*327)}{80+327} = 11.7$ ms.

This formula is extensible to any number of elements by summing the products of all request rates and their associated latencies, then dividing by the aggregate of all request rates.

An example of the average workload and performance metrics for the first 17 volumes of the virtual disks is shown in the following example. Each line represents the average of the full data capture period, which enables you to generally compare long-term virtual disk performance metrics and quickly determine which volumes to focus on for further analysis.

Example of virtual disk performance summary over the data capture duration



ID	Read Hit Req/s	Read Hit MB/s	Read Hit Latency (ms)	Read Miss Req/s	Read Miss MB/s	Read Miss Latency (ms)	Write Req/s	Write MB/s	Write Latency (ms)	Flush MB/s	Mirror MB/s	Prefetch MB/s
0	0	0.0	0.0	71	0.6	27.2	49	0.4	18.5	0.4	1.0	0.0
1	0	0.0	0.0	19	0.3	15.8	0	0.0	0.0	0.0	0.0	0.0
2	0	0.0	0.0	59	3.9	12.3	0	0.0	0.0	0.0	3.9	3.9
3	0	0.0	0.0	0	0.0	0.0	61	4.0	11.9	4.0	4.0	0.0
4	0	0.0	0.0	0	0.0	0.0	48	3.2	15.3	3.2	3.2	0.0
5	0	0.0	0.0	72	0.6	15.3	49	0.4	15.9	0.4	0.4	0.0
6	0	0.0	0.0	19	0.3	26.3	0	0.0	0.0	0.0	0.3	0.0
7	233	15.3	2.7	26	1.7	2.6	0	0.0	0.0	0.0	0.0	15.3
8	0	0.0	0.0	72	0.6	14.7	49	0.4	8.9	0.4	0.4	0.0
9	0	0.0	0.0	19	0.3	17.3	0	0.0	0.0	0.0	0.0	0.0
10	408	26.8	1.6	0	0.0	0.0	0	0.0	0.0	0.0	0.0	26.8
11	0	0.0	0.0	0	0.0	0.0	93	6.1	7.8	6.1	6.1	0.0
12	0	0.0	0.0	0	0.0	0.0	110	7.2	6.6	7.2	7.2	0.0
13	0	0.0	0.0	72	0.6	16.5	49	0.4	10.4	0.4	0.4	0.0
14	0	0.0	0.0	19	0.3	15.0	0	0.0	0.0	0.0	0.0	0.0
15	259	17.0	2.7	0	0.0	0.0	0	0.0	0.0	0.0	0.0	17.0
16	0	0.0	0.0	72	0.6	14.4	49	0.4	8.9	0.4	0.4	0.0

Physical disk statistics

In an EVA, usually the value of the physical disk information is providing detailed performance analysis at the disk group level. Because there are so many physical disks in an EVA system, where all disks are members of virtual disks, it is difficult to extract meaningful information on an individual physical disk basis. However, if there are different capacity physical disks or different speed disks within a single disk group, individual or drive subgroup reporting might be necessary. This is actually accomplished easily in HP Command View EVAPerf using the CLI `pda` command to view overall data rates in a physical matrix of physical disks by shelf and bay. However, queue depths, request rates, or latencies must be done manually.

To obtain performance data on the physical disks, rearrange the data for individual physical disks into time series. Again, each controller presents its statistics separately.

The following example shows the contribution from each controller, but usually it is preferable to combine controller statistics for each physical disk.

Physical disk performance data extracted from the output data file

```
Time,ID,Drive Queue Depth,Read Req/s,Read MB/s,Read Latency (ms),Write Req/s,Write
MB/s,Write Latency (ms),Enc,Bay,Grp ID,Ctlr
11:41:00,0,57,0,0.08,18.8,0,0.03,31.3,2,1,256,A
11:49:26,0,174,10,0.93,16.1,5,0.59,27.6,2,1,256,A
11:51:27,0,168,11,0.99,16.9,8,0.55,22.1,2,1,256,A
11:53:28,0,170,11,1,16,8,0.57,21.7,2,1,256,A
…
11:41:00,0,95,0,0.07,13.3,0,0.19,30.4,2,1,256,B
11:49:26,0,151,12,0.92,16.6,10,0.85,21.8,2,1,256,B
11:51:27,0,149,11,0.92,17.4,12,0.83,20.1,2,1,256,B
11:53:28,0,146,11,0.9,16.7,12,0.8,20.3,2,1,256,B
…
11:41:00,1,50,0,0.06,14.1,0,0.03,31.7,3,1,256,A
11:49:26,1,187,11,1.05,18.2,5,0.59,28.3,3,1,256,A
11:51:27,1,166,11,0.99,15.7,8,0.55,21.5,3,1,256,A
11:53:28,1,173,11,1,16,8,0.57,22.5,3,1,256,A
…
11:41:00,1,88,0,0.07,13.9,0,0.19,33.7,3,1,256,B
11:49:26,1,156,11,0.93,17.2,10,0.85,23.2,3,1,256,B
11:51:27,1,146,11,0.91,16,12,0.83,20.9,3,1,256,B
11:53:28,1,146,12,0.95,16.3,12,0.8,20.6,3,1,256,B
…
```

The previous example shows the first few samples for physical disks 0 and 1 for both controllers. The next step would be to combine statistics from both controllers for each physical disk so that all activity is coalesced for each line. Further reduction is discussed in the next section.

To identify physical disks that have might have disproportionate levels of workload skew, average all request rates and data rates for each physical disk separately for the measurement interval. Latencies must be combined using the weighted average formula.

HP Continuous Access tunnel statistics

To obtain performance data on the HP Continuous Access tunnels, separate all tunnel records into separate time series files, which will show all active and inactive tunnel statistics in a contiguous file.

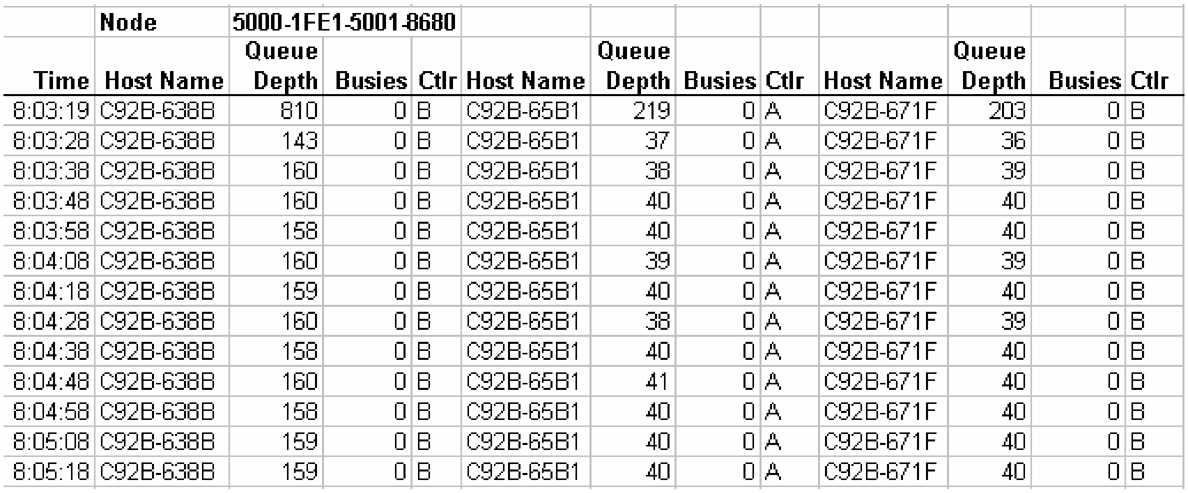
Single HP Continuous Access tunnel statistics extracted from the output data file (This should be done for each active tunnel, which is usually a subset of all open tunnels.)

```
Time,Ctlr,Tunnel Number,Host Port,Source ID,Dest. ID,Round Trip Delay (ms),Copy
Retries,Write Retries,Copy In MB/s,Copy Out MB/s,Write In MB/s,Write Out MB/s,Node
9:39:39,B,3,3,40800,30700,12.8,0,0,0,0,0,16.14,5000-1FE1-5001-8680
9:39:48,B,3,3,40800,30700,12.8,0,0,0,0,0,14.87,5000-1FE1-5001-8680
9:39:58,B,3,3,40800,30700,12.8,0,0,0,0,0,16.03,5000-1FE1-5001-8680
9:40:08,B,3,3,40800,30700,12.8,0,0,0,0,0,16.14,5000-1FE1-5001-8680
9:40:18,B,3,3,40800,30700,12.8,0,0,0,0,0,16.27,5000-1FE1-5001-8680
9:40:28,B,3,3,40800,30700,12.8,0,0,0,0,0,16.07,5000-1FE1-5001-8680
```

Host connection statistics

To obtain performance data on the host connection, separate queue depth for each host connection into time series file, one host connection per field.

Host connection statistics arranged as a time series (In this example, three host connections are shown on a one line for each timestamp.)



	Node	**5000-1FE1-5001-8680**										
Time	**Host Name**	**Queue Depth**	**Busies**	**Ctlr**	**Host Name**	**Queue Depth**	**Busies**	**Ctlr**	**Host Name**	**Queue Depth**	**Busies**	**Ctlr**
8:03:19	C92B-638B	810	0	B	C92B-65B1	219	0	A	C92B-671F	203	0	B
8:03:28	C92B-638B	143	0	B	C92B-65B1	37	0	A	C92B-671F	36	0	B
8:03:38	C92B-638B	160	0	B	C92B-65B1	38	0	A	C92B-671F	39	0	B
8:03:48	C92B-638B	160	0	B	C92B-65B1	40	0	A	C92B-671F	40	0	B
8:03:58	C92B-638B	158	0	B	C92B-65B1	40	0	A	C92B-671F	40	0	B
8:04:08	C92B-638B	160	0	B	C92B-65B1	39	0	A	C92B-671F	39	0	B
8:04:18	C92B-638B	159	0	B	C92B-65B1	40	0	A	C92B-671F	40	0	B
8:04:28	C92B-638B	160	0	B	C92B-65B1	38	0	A	C92B-671F	39	0	B
8:04:38	C92B-638B	158	0	B	C92B-65B1	40	0	A	C92B-671F	40	0	B
8:04:48	C92B-638B	160	0	B	C92B-65B1	41	0	A	C92B-671F	40	0	B
8:04:58	C92B-638B	158	0	B	C92B-65B1	40	0	A	C92B-671F	40	0	B
8:05:08	C92B-638B	159	0	B	C92B-65B1	40	0	A	C92B-671F	40	0	B
8:05:18	C92B-638B	159	0	B	C92B-65B1	40	0	A	C92B-671F	40	0	B

Virtual disk by disk group statistics

The vdg command line option is valuable for capturing controller workloads. This option provides the aggregate virtual disk statistics on a disk group basis. If there is a single disk group behind the controller, then the metrics exactly correlate to controller statistics. Otherwise, the metrics must be summed over all disk groups.

Virtual disk statistics by disk group (The first two systems each have a single disk group, so the data presented is exactly the controller workload. The third system has three disk groups, so controller workloads must be acquired through summation of each disk group's data.)

```
Disk     Total  Total  Average Total  Total  Average  Total  Total  Average  Total  Total  Total    Ctlr   Node
Group    Read   Read   Read    Read   Read   Read     Write  Write  Write    Flush  Mirror Prefetch
         Hit    Hit    Hit     Miss   Miss   Miss     Req/s  MB/s   Latency  MB/s   MB/s   MB/s
         Req/s  MB/s   Latency Req/s  MB/s   Latency         (ms)
                       (ms)                  (ms)
-------  -----  -----  ------- -----  -----  -------  -----  -----  -------  -----  ------ -------- ----   ----

Default    0    0.00    0.0    7963   65.37   23.8    5458   44.84   17.1    44.78  44.85   0.00    2003   Cab2
Default    0    0.00    0.0       0    0.00    0.0       0    0.00    0.0     0.00   0.00   0.00    200D   Cab2
Default    0    0.00    0.0       0    0.00    0.0    5453   44.84    0.7    44.86  44.84   0.00    301Q   Cab4
Default    0    0.00    0.0       0    0.00    0.0       0    0.00    0.0     0.00   0.00   0.00    Z001   Cab4
    DG2    1    0.01    0.4    2543   20.84   11.8    1733   14.21    0.7    13.71  15.81   0.00    T03T   Cab6
    DG2    1    0.01    0.1    1351   11.07   10.9     892    7.31    0.9     6.75  14.05   0.00    T07C   Cab6
    DG3    1    0.01    0.2    2323   19.03   13.1    1547   12.68    0.4    12.80   0.00   0.00    T03T   Cab6
    DG3    1    0.01    0.2    2103   17.23   14.5    1358   11.13    0.5    11.62   0.00   0.00    T07C   Cab6
Default    0    0.00    0.0    1345   11.02   11.2     935    7.67    0.4     7.90   0.00   0.00    T03T   Cab6
Default    0    0.00    0.0    2775   22.75   10.9    1853   15.19    0.4    15.09   0.00   0.00    T07C   Cab6
```

Recombining for more information

The time series files can further be manipulated to combine specific object counters to pinpoint performance data at the controller and disk group level.

Storage subsystem

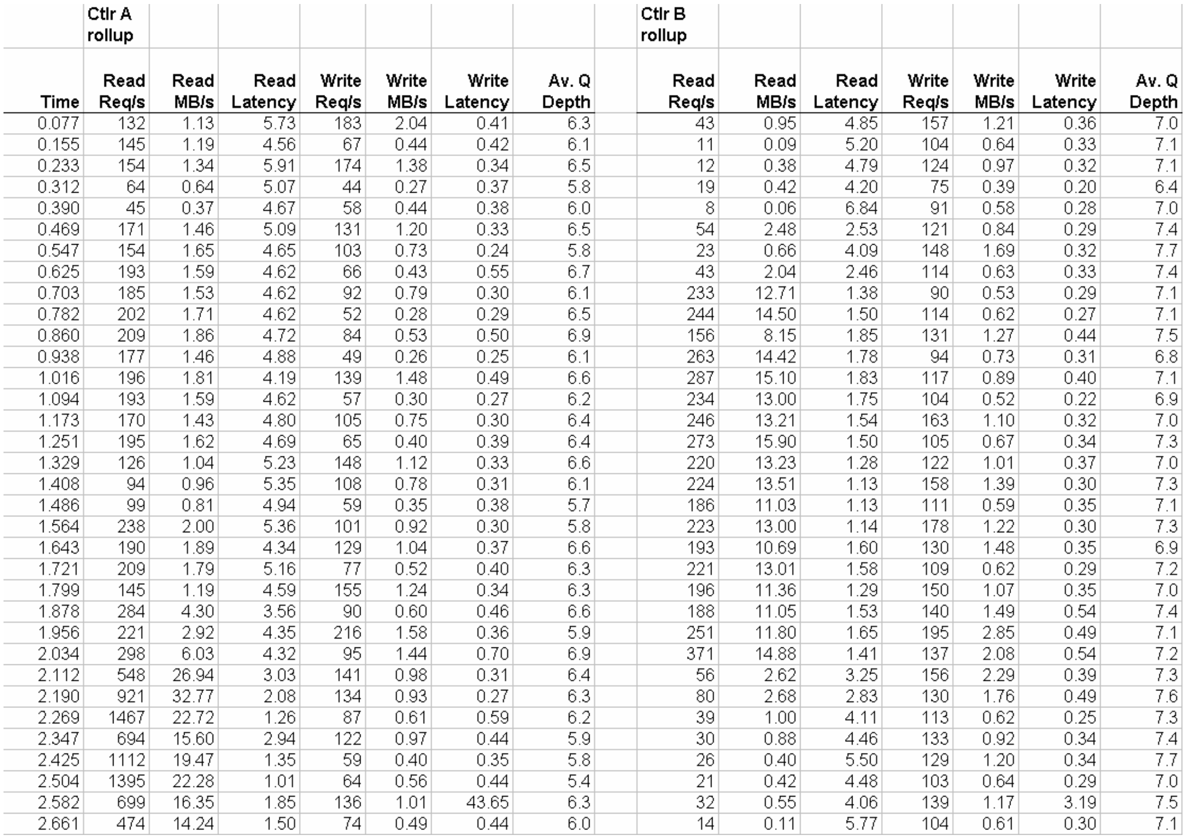
To obtain detailed information on controller activity and storage subsystem activity, recombine the host port data using the following procedure.

1. Beginning with the host port data grouped by timestamp, combine all host port traffic for each controller.

   This will display the controller workload and performance information, including the controller read and write latency, which have been combined from the individual host port statistics and must be calculated as part of this step.

2. Combine the latencies as weighted averages using the formula on page 23.

Recombined host port statistics provide controller-level performance data

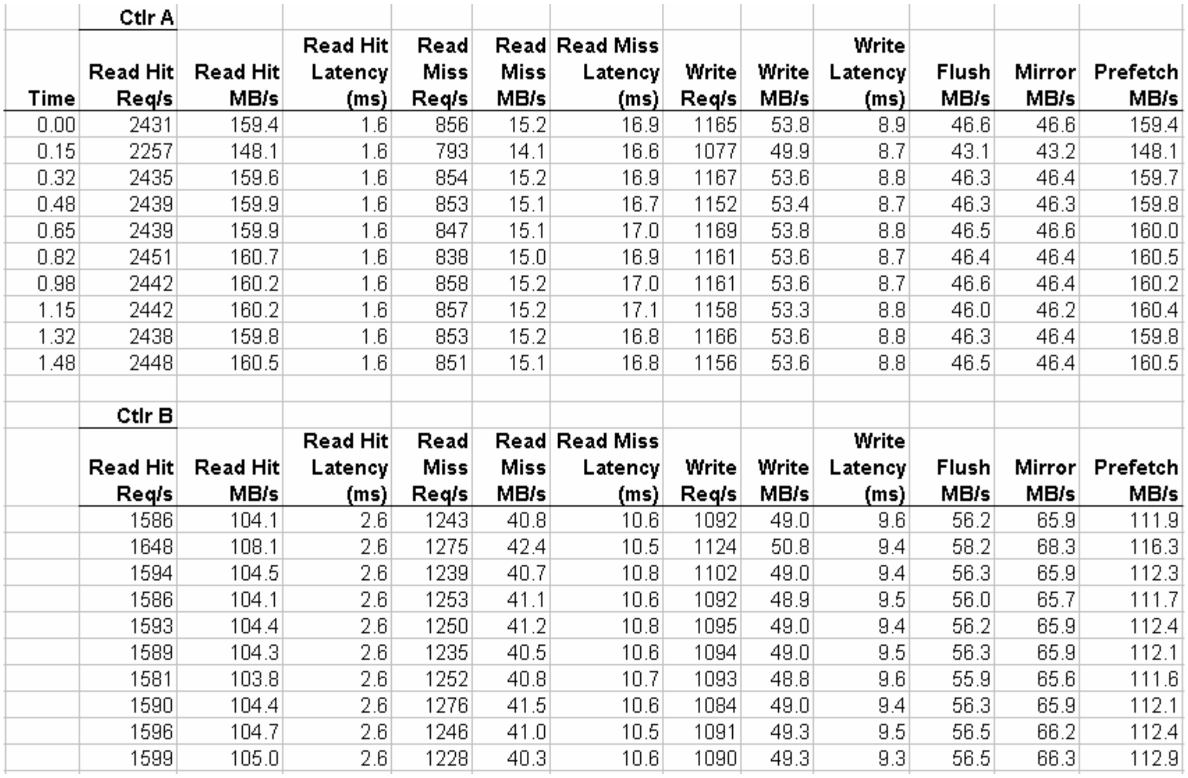


	Ctlr A rollup							Ctlr B rollup						
Time	Read Req/s	Read MB/s	Read Latency	Write Req/s	Write MB/s	Write Latency	Av. Q Depth	Read Req/s	Read MB/s	Read Latency	Write Req/s	Write MB/s	Write Latency	Av. Q Depth
0.077	132	1.13	5.73	183	2.04	0.41	6.3	43	0.95	4.85	157	1.21	0.36	7.0
0.155	145	1.19	4.56	67	0.44	0.42	6.1	11	0.09	5.20	104	0.64	0.33	7.1
0.233	154	1.34	5.91	174	1.38	0.34	6.5	12	0.38	4.79	124	0.97	0.32	7.1
0.312	64	0.64	5.07	44	0.27	0.37	5.8	19	0.42	4.20	75	0.39	0.20	6.4
0.390	45	0.37	4.67	58	0.44	0.38	6.0	8	0.06	6.84	91	0.58	0.28	7.0
0.469	171	1.46	5.09	131	1.20	0.33	6.5	54	2.48	2.53	121	0.84	0.29	7.4
0.547	154	1.65	4.65	103	0.73	0.24	5.8	23	0.66	4.09	148	1.69	0.32	7.7
0.625	193	1.59	4.62	66	0.43	0.55	6.7	43	2.04	2.46	114	0.63	0.33	7.4
0.703	185	1.53	4.62	92	0.79	0.30	6.1	233	12.71	1.38	90	0.53	0.29	7.1
0.782	202	1.71	4.62	52	0.28	0.29	6.5	244	14.50	1.50	114	0.62	0.27	7.1
0.860	209	1.86	4.72	84	0.53	0.50	6.9	156	8.15	1.85	131	1.27	0.44	7.5
0.938	177	1.46	4.88	49	0.26	0.25	6.1	263	14.42	1.78	94	0.73	0.31	6.8
1.016	196	1.81	4.19	139	1.48	0.49	6.6	287	15.10	1.83	117	0.89	0.40	7.1
1.094	193	1.59	4.62	57	0.30	0.27	6.2	234	13.00	1.75	104	0.52	0.22	6.9
1.173	170	1.43	4.80	105	0.75	0.30	6.4	246	13.21	1.54	163	1.10	0.32	7.0
1.251	195	1.62	4.69	65	0.40	0.39	6.4	273	15.90	1.50	105	0.67	0.34	7.3
1.329	126	1.04	5.23	148	1.12	0.33	6.6	220	13.23	1.28	122	1.01	0.37	7.0
1.408	94	0.96	5.35	108	0.78	0.31	6.1	224	13.51	1.13	158	1.39	0.30	7.3
1.486	99	0.81	4.94	59	0.35	0.38	5.7	186	11.03	1.13	111	0.59	0.35	7.1
1.564	238	2.00	5.36	101	0.92	0.30	5.8	223	13.00	1.14	178	1.22	0.30	7.3
1.643	190	1.89	4.34	129	1.04	0.37	6.6	193	10.69	1.60	130	1.48	0.35	6.9
1.721	209	1.79	5.16	77	0.52	0.40	6.3	221	13.01	1.58	109	0.62	0.29	7.2
1.799	145	1.19	4.59	155	1.24	0.34	6.3	196	11.36	1.29	150	1.07	0.35	7.0
1.878	284	4.30	3.56	90	0.60	0.46	6.6	188	11.05	1.53	140	1.49	0.54	7.4
1.956	221	2.92	4.35	216	1.58	0.36	5.9	251	11.80	1.65	195	2.85	0.49	7.1
2.034	298	6.03	4.32	95	1.44	0.70	6.9	371	14.88	1.41	137	2.08	0.54	7.2
2.112	548	26.94	3.03	141	0.98	0.31	6.4	56	2.62	3.25	156	2.29	0.39	7.3
2.190	921	32.77	2.08	134	0.93	0.27	6.3	80	2.68	2.83	130	1.76	0.49	7.6
2.269	1467	22.72	1.26	87	0.61	0.59	6.2	39	1.00	4.11	113	0.62	0.25	7.3
2.347	694	15.60	2.94	122	0.97	0.44	5.9	30	0.88	4.46	133	0.92	0.34	7.4
2.425	1112	19.47	1.35	59	0.40	0.35	5.8	26	0.40	5.50	129	1.20	0.34	7.7
2.504	1395	22.28	1.01	64	0.56	0.44	5.4	21	0.42	4.48	103	0.64	0.29	7.0
2.582	699	16.35	1.85	136	1.01	43.65	6.3	32	0.55	4.06	139	1.17	3.19	7.5
2.661	474	14.24	1.50	74	0.49	0.44	6.0	14	0.11	5.77	104	0.61	0.30	7.1

3. To acquire a more complete performance picture for the controllers, combine all virtual disk statistics for each timestamp by controller as shown in the following example. This provides back-end and front-end data and separates read hits from misses. The request rates, data rates, and latencies should agree quite closely with the host port rollups for the same timestamps.

Timestamps are shown only with controller A because the two tables would normally be shown beside each other. In this example, the timestamps have been modified to show elapsed time in minutes instead of absolute wall-clock time.

Recombined virtual disk statistics to provide controller-level performance data



	Ctlr A											
Time	Read Hit Req/s	Read Hit MB/s	Read Hit Latency (ms)	Read Miss Req/s	Read Miss MB/s	Read Miss Latency (ms)	Write Req/s	Write MB/s	Write Latency (ms)	Flush MB/s	Mirror MB/s	Prefetch MB/s
0.00	2431	159.4	1.6	856	15.2	16.9	1165	53.8	8.9	46.6	46.6	159.4
0.15	2257	148.1	1.6	793	14.1	16.6	1077	49.9	8.7	43.1	43.2	148.1
0.32	2435	159.6	1.6	854	15.2	16.9	1167	53.6	8.8	46.3	46.4	159.7
0.48	2439	159.9	1.6	853	15.1	16.7	1152	53.4	8.7	46.3	46.3	159.8
0.65	2439	159.9	1.6	847	15.1	17.0	1169	53.8	8.8	46.5	46.6	160.0
0.82	2451	160.7	1.6	838	15.0	16.9	1161	53.6	8.7	46.4	46.4	160.5
0.98	2442	160.2	1.6	858	15.2	17.0	1161	53.6	8.7	46.6	46.4	160.2
1.15	2442	160.2	1.6	857	15.2	17.1	1158	53.3	8.8	46.0	46.2	160.4
1.32	2438	159.8	1.6	853	15.2	16.8	1166	53.6	8.8	46.3	46.4	159.8
1.48	2448	160.5	1.6	851	15.1	16.8	1156	53.6	8.8	46.5	46.4	160.5
	Ctlr B											
	Read Hit Req/s	Read Hit MB/s	Read Hit Latency (ms)	Read Miss Req/s	Read Miss MB/s	Read Miss Latency (ms)	Write Req/s	Write MB/s	Write Latency (ms)	Flush MB/s	Mirror MB/s	Prefetch MB/s
	1586	104.1	2.6	1243	40.8	10.6	1092	49.0	9.6	56.2	65.9	111.9
	1648	108.1	2.6	1275	42.4	10.5	1124	50.8	9.4	58.2	68.3	116.3
	1594	104.5	2.6	1239	40.7	10.8	1102	49.0	9.4	56.3	65.9	112.3
	1586	104.1	2.6	1253	41.1	10.6	1092	48.9	9.5	56.0	65.7	111.7
	1593	104.4	2.6	1250	41.2	10.8	1095	49.0	9.4	56.2	65.9	112.4
	1589	104.3	2.6	1235	40.5	10.6	1094	49.0	9.5	56.3	65.9	112.1
	1581	103.8	2.6	1252	40.8	10.7	1093	48.8	9.6	55.9	65.6	111.6
	1590	104.4	2.6	1276	41.5	10.6	1084	49.0	9.4	56.3	65.9	112.1
	1596	104.7	2.6	1246	41.0	10.5	1091	49.3	9.5	56.5	66.2	112.4
	1599	105.0	2.6	1228	40.3	10.6	1090	49.3	9.3	56.5	66.3	112.9

4. If desired, combine these performance matrices further to acquire a single table of array-level performance information.
5. Graph the controller-level data over the time interval of interest.

Disk groups

To determine how heavily the disks are being exercised, it is easier to view the physical disk data aggregated into disk groups, which enables you to determine whether the disk group size is sufficient for the workload. This is most easily accomplished using the CLI `pdg` command, which groups all activity from all physical disks according to disk group.

The following example is the output from a single EVA3000 or EVA5000 system with three disk groups. This plain text shows the activity for all disk groups on the system and for each controller providing the workload.

There are two opportunities for extracting additional information. First, in this example, all workload is from a single controller, but because that is not always the case, it would be beneficial to combine workloads to view the full workload to a disk group. Secondly, in the continuous mode, all information for all counters is presented for each sample interval. To view each disk group as a time series, it would be necessary to separate the information for each disk group into separate files or separate fields on one line for each timestamp.

If it is necessary to view the workload separately for drive subgroups, the entire physical disk data must be processed to extract data accordingly.

Physical disk performance data by disk group

```
Disk   Average Average Average Average Average Average  Average  Average  Number Ctlr Node
Group Drive   Drive   Read    Read    Read    Write    Write    Write    of
      Queue   Latency Req/s   MB/s    Latency Req/s    MB/s     Latency  Disks
      Depth   (ms)                    (ms)                      (ms)
-------------------------------------------------------------------------------------------
01     2       10.1    0       0.00      -     81       1.34      -       48      A   …-AF10
01     0        0.0    0       0.00      -      0       0.00      -       48      B   …-AF10
02     2        8.6    0       0.00      -     51       0.84      -       38      A   …-AF10
02     0        0.0    0       0.00      -      0       0.00      -       38      B   …-AF10
Default 2      11.9    0       0.00      -     48       0.79      -       80      A   …-AF10
Default 0       0.0    0       0.00      -      0       0.00      -       80      B   …-AF10
```

Performance problems

The first step is to determine if there is indeed a performance problem. This is done from a user's requirements perspective. It is not unusual for a storage user to inspect performance statistics and decide that a certain metric is "bad." While some metrics might readily indicate a problem, a judgment of goodness must be made at the system or user level. Some examples are: "My database application is getting storage access time-outs," or "My backup window used to be fine, but now I'm impacting production," or "Users are complaining of long response times on their e-mail." Examples of non-problems are: "These host port queue depths are way too high," or "HP Command View EVAPerf shows lots of read activity on physical drives, but I'm doing only writes to my Vraid 5 volumes." It is far more helpful to define a performance problem in terms of its effect on the user instead of interpretation of the metrics.

If possible, begin with host-based metrics that quantify the problem, then compare with controller performance metrics. If the host-based metrics that indicate a problem are consistently different than that seen from the controller metrics, the problem is likely not in the storage system. Examples of non-storage related poor performance causes are:

- Host problems with the application. While addressing those problems is well beyond the scope of this paper, there are several sources for system performance and tuning for many applications and their environments.
- Too few paths to the storage system or paths poorly configured. Current host bus adapters can deliver on the order of 25 to 30,000 IO/s and 200 MB/s for reads. Driver queue depths, I/O response coalescing, link speed settings, and so on can have a profound effect on storage performance. Each adapter can have impressive performance, but if the SAN fabric is misconfigured, there can still be a performance problem.
- Operating system configuration parameters such as maximum transfer size, storage access restrictions, file system parameters, page swapping, and so on.

Reviewing the data

Performance problems can sometimes be identified using the HP Command View EVAPerf text output data as it is without further processing. A lengthy time series is not always necessary to identify clear problems such as excessive port imbalances or saturation, high response times at the virtual disk level, heavy proxy controller workloads, or unexpectedly finding virtual disks in write-through mode. It is critical, however, to be mindful that workloads are usually quite dynamic and variable, so what is observed at one time instance might not be representative of the workload character.

A reasonable alternative to examining top level output data is to monitor the same counters in the Perfmon window, which has the added benefit of being able to select specific objects and counters within the performance object. Not all EVA counters are available in Perfmon, but most essential ones are. The Perfmon histogram function enables you to see the time varying metrics in a stationary position, so short-term direct comparison is effective.

At every level of analysis, determine if the reported behavior conforms to what is expected. Awareness of reasonable values can facilitate closing in on a performance problem. There are many factors to consider, a few of which are:

- Server activity—Are the appropriate host adapters providing expected relative contributions to workload?
- Virtual disk workload—Are the workloads to each virtual disk appropriate? For example, are sequential reads or random read/write mix expected? Are transfer sizes in line with the given workload? Does workload variation correspond to known changes in usage patterns? Is each virtual disk active at the appropriate time?
- HP Continuous Access tunnel activity—Do Write Out/In values correlate to the corresponding remote mirror write activity? Are copies active when they should not be or not active when they should be? Do ping times reflect expected link parameters and behavior?

Host port

The first look is often at the host port information. This provides a top level view of what is going into and out of the array, how port workload is balanced, and if the metrics correspond to the user’s observations. Verify latencies, data rates, request rates, and queue depths with respect to externally generated performance metrics if performance problem identification began there. Those metrics, including those from server performance tools, database internal metrics, fabric monitors, and so on, all can indicate problems with the storage array. It is easiest to confirm the problem at this level.

Single port data is usually not as informative as the aggregate across the ports for the controller load. A single port has a fixed upper limit of about 200 MB/s based on 2-Gb Fibre Channel protocol, so data rates close to that might be experiencing the technology limit. Otherwise, the port (link) is not the bottleneck.

Estimate the controller load from the aggregate of all ports on that controller. In absolute terms, determine the level of activity compared to practical performance limits for the given workload. Determine controller workload imbalance. If the absolute levels are well below the controller capability, any imbalance should not be a problem. If imbalance is significant and one controller is approaching some limit, it indicates a rebalancing opportunity. While the rebalancing concept is quite straightforward, it will be anywhere from simple to do to impossible to perform, depending on constraints such as fabric topology, ISL constraints, server workload skew, and virtual disk workload imbalance.

Controller statistics

Controller utilizations will show how hard the controller processor is working. If the processor utilization is low, yet there is a performance problem with the array, then the physical disk storage might be the bottleneck. Also, high data rate sequential workloads can saturate controller data paths yet leave the controller CPU with low utilization. It is not unusual for CPU utilization and data rates to be inversely related.

The data path utilization reported is how much of the controller processor time is allocated to servicing data requests. The difference between CPU utilization and data utilization is the overhead the controller is allocating to non-client data transfer activities, such as external communication, sparing and leveling, and virtual volume management. Some of these activities might not be a priority and can be interrupted by data transfer requests from a client, which allows opportunity for further host activity, despite the appearance of a very busy controller.

It is important to realize that there are two critical resources within the controller—the processor itself and the data path, which includes elements such as bus bandwidth, cache bandwidth, and fiber port parameters. The processor manages all activity and communication in the controller, and the data path moves the data. Usually, the processor will be the critical performance element for small transfer sizes and high request rates, while for large transfer sizes, the data path from server through controller to physical disk will have a greater impact on performance.

Host connection

Host connection queue depth is a measure of the average number of requests in process for each host adapter. This value is not so much a performance metric as it is an indication of how much load each host is offering to the array. The actual values will be workload-dependent. A quick look here can verify appropriate server loading balance. High queue depths that are deemed problematic should be traced to the virtual volumes and the back end systems that support those volumes.

The presence of busies in the host connection metrics are usually an indication of a performance problem. Each one represents a signal from the controller to the host to suspend further requests until some internal queue in the controller is sufficiently reduced. Queue full responses are grouped in this metric and whichever is issued is operating system dependent. The subsequent server response to a busy or queue full signal is also operating system dependent.

Virtual disk

Workloads are most thoroughly defined, and the interaction between workloads most easily shown, at the virtual disk level.

Read hit and read miss statistics are reported separately. Because transaction workloads such as database queries and updates usually have a random data access range that far exceeds the read cache capacity, those workloads will see very few opportunities to access data already in cache. Therefore, read hits generally occur primarily for sequential read workloads where the sequential detect and prefetch algorithms are highly effective. This can be confirmed by comparing virtual disk read hit data rates with their corresponding prefetch data rates. Read hits usually have low, often sub millisecond, request latencies.

In write-back mode, writes are always accepted by cache, the request is acknowledged as complete, and the data marked for flush to the back end. For this scenario, write latencies are always low, sometimes sub millisecond, but usually less than a few milliseconds. When the write workload (or other back-end activity) reaches a level such that cache flushes cannot keep pace with the front-end write demand, the write request must wait for cache space before it can be completed, and write latencies increase, sometimes significantly. This can also happen when the mirror port utilization becomes very high and cannot copy the data to the other controller in time to support the workload. In either case, the request is not marked as complete until the data is in cache in both controllers.

Write latencies can increase unexpectedly on one controller for the battery condition presented in the "Wr Mode" description under the "**Error! Reference source not found.**" section on page **Error! Bookmark not defined.**. In this case, there will be a controller event logged and viewable in HP StorageWorks Command View.

Latencies should be appropriate for the workload. Read latencies less than 15 ms should be acceptable for most random workloads. Write latencies of a few milliseconds indicate that writes to the controller are below its flush capability. Double-digit write latencies might still be appropriate, depending on server application sensitivities, and indicate either the virtual disk is in write-through mode (unlikely) or the cache flush mechanism is approaching its limits (more likely) or high utilization on the mirror port. Large transfer sequential workloads often can have higher latencies.

Virtual disk latencies are not independent of each other. Cache flush operations and mirror port utilization depend only on the aggregate workload for the controller. Thus, one virtual disk's workload can significantly impact another's latencies.

Mirror ports are active for the owning controller when a write request is received and cache mirroring is enabled for that virtual disk or when a read request is received by a proxy controller for that virtual disk. Because mirror port data rates are always the write activity (outbound) for each controller for the associated virtual disk, the reported mirror port activity to a given virtual disk is the sum of write data to that virtual disk and read activity in support of its proxy counterpart. When the volume being written to is a Vraid 5 volume and sequential activity is not detected, then the mirror port must carry twice as much data as that written to the host port because parity must be transferred with the data to ensure data integrity through all failure scenarios.

Flush data rates represent the movement of data from cache to physical disk. For each controller, this will be the aggregate of all data written to any virtual disk owned by that controller. A proxy controller will not flush cache for any host write requests. Neither parity nor mirror redundancy data is included in the flush rates.

Cache flush is asynchronous with respect to front-end writes, so if data is sampled at short intervals, the data rates from host writes and flush might not agree, especially for low write workloads.

Large transfer sequential workloads can negatively impact random workload response times on a different virtual disk. All virtual disks share storage system resources. The sequential workload can easily drive up internal bus utilization, and large transfers will take relatively long intervals to complete. This condition forces queuing for internal system resources, which can drive up unrelated, random small transfer latencies.

Behavior can be markedly different for different RAID levels. Vraid1 forces duplicate writes, one for each member of the mirror pair. Because these writes happen in parallel and asynchronously, the write performance will be close to Vraid 0. Reads will be shared between the two mirror volumes, so there is the potential for Vraid 1 read performance to exceed Vraid 0. Vraid 5 requires parity management, which will behave differently under different conditions. Sequential writes to Vraid 5 volumes will have a minor parity impact, but non-sequential workloads will endure far more expensive back-end activity. The same effect is seen on mirror traffic for the volume.

HP Continuous Access EVA source volumes will almost always be impacted by replication activity. For synchronous transfer mode, a write request cannot be completed until the data from the request is stored on the replication system and an acknowledgment returned. Link speeds and link delay will impact this process. Asynchronous mode will benefit light workloads but will become less beneficial as the write workload to a source volume becomes more intense. When the internal link queue is full, the replication mode essentially reverts to synchronous mode.

Proxy access to a virtual disk can have a negative impact with high workload intensity. Reads from a virtual disk through the proxy controller still require the owning controller to retrieve the information, and it uses a mirror port to transfer the data between controllers. This condition results in elongated read response times and higher mirror port utilization. Writes to either controller require mirror transfers anyway, but the owning controller must still flush the data. This might not be a problem, but balancing the workload across controllers will require care if proxy workloads are involved.

Physical disk

It is usually difficult to extract good information from the raw physical disk output data because of the enormity of the data and short-term variability within a disk group. However, a visual scan might be effective in identifying a disk or subset of disks that have relatively high queue depths and latencies in sample after sample. Typically, it takes more aggressive post processing to glean useful information from the physical disk data.

All virtual disk storage is physically allocated across all physical disks in the corresponding disk group, so all physical drives will share in all virtual disk activity. There is no association between virtual disks and physical disks in a single disk group. Thus, there should be essentially uniform activity across all equal size drives in a disk group when averaged over longer intervals.

There are conditions under which there can be long-term workload skew to individual drives. The first is when there are disks of unequal capacity within the disk group. The larger disks will have proportionately larger I/O workloads because EVA allocates storage in proportion to the disk capacity. For example, on a system with 36-GB disks and 146-GB disks, you will see an average of four times the workload on the 146-GB disks than for the 36-GB disks. In cases such as this, it would be helpful to examine physical disk subgroups of equal size drives

Reported physical disk activity includes everything that happens behind the controllers, including parity and mirror redundancy writes as well as parity and data reads for parity calculations. It also includes background activity such as metadata management, sparing and leveling when appropriate, and all I/O traffic to support Snapshots, Snapclones, and HP Continuous Access traffic. Mapping physical disk activity to front-end activity ranges from difficult to impossible.

HP Continuous Access tunnel

Look first at the round trip delay values (ping times) for active tunnels. These should be in line with what is expected for the link at the given time. Low values would be expected for local links with relatively light write workloads to the source volumes. Higher values would be expected for heavy write traffic or for ISLs with long intrinsic delay times. These times should vary mostly with load. Delays should be essentially symmetrical as reported by the source and destination systems.

Copy Out data should be 0 for all volumes that are fully normalized between source and destination sites. Non-zero data rates here indicate that normalization is not complete or a new copy has been triggered by a previous link failure and subsequent restoration or by full copy request.

Write Out data rates include all write traffic to the corresponding remote mirror volumes, any write retries, and merge data from a HP Continuous Access EVA log on the source side. Merge data is "catch up" data from a temporary link disruption. Comparing this value to the write data for all source volumes, along with ping time, provides a first look at ISL health.

Copy In and Copy Out values, as well as Write In and Write Out values, should be essentially the same except for deviations caused by short sample intervals or other timing anomalies. Copy values will be 0 for all HP Continuous Access EVA groups in which the member volumes are fully normalized, unless a link disruption has triggered a full copy.

The parameters and behavior of the inter-site link will have a non-trivial effect on performance. A low speed link can limit the rate at which data is transferred to the replication system. If the write workload demand is greater than that rate, the link becomes the bottleneck. Bursts of write activity can result in high latencies even if the long-term link utilization is low. The EVA controller architecture provides far better burst capability than the link itself, so it will experience high queuing for HP Continuous Access tunnel resources with slower links and highly intermittent workloads. This in turn is reflected directly to virtual disk latencies.

Controller workload

To analyze controller workload, it is necessary to combine host port data or virtual disk data to acquire all of the relevant information on each controller. The former is easy to do either directly with a post processing tool or by reorganizing port data as described on page 27 and then summing the data for the corresponding controller. Extracting controller level data from a virtual disk provides a richer set of information. If there is a single disk group in the target EVA, controller information can be viewed directly using the `vdg` command.

Both front-end and back-end activity should be reviewed in a virtual disk rollup per controller. The front-end information should correlate well with the host port rollup for request rates, data rates, and latencies. Read hits and misses will be tallied separately for the virtual disk rollup but could be easily combined by the processing tool.

Front-end activity is a clear and detailed picture of the workload presented by all servers attached to the storage array and its response to that workload. Workload balance between controllers is clearly shown and charted over time if the time series data reduction is done. If transfer sizes are calculated, that can add supporting information for sequential and random workload identification.

Back-end activity includes prefetch, mirror, and flush data rates. If EVA4000, EVA6000, and EVA8000 mirror rates are high compared to write rates for each controller, it could indicate a high proxy read activity. If mirror rates are low compared with write rates, it could indicate write workload to volumes that have cache mirroring disabled.

Flush rates are designed to be adaptive based on the write workload intensity. A time series for a system with light write workloads will show intermittent flush activity. As the total write activity to the controller increases, which would include activity such as write-ins or copy-ins from an HP Continuous Access EVA source system or writes on behalf of the proxy controller, flush rates will become higher and steadier. When flush rates are consistently equal to the aggregate write rates, the maximum write rate to that controller has probably been achieved. At that point, write latencies might increase significantly.

Prefetch rates should correlate well with read hit rates.

Controller balance should be a concern if one controller is nearing the upper limit for the current workloads. This is more often a consideration for data rates, as opposed to requests per second, because it is much easier to drive the internal controller buses into saturation than to exceed the capability of a controller to process requests. One configuration in which controller balance can be an issue is with an HP Continuous Access EVA implementation. Because all volumes in a single group must be presented to the same controller, all activity for those virtual disks, including replication traffic, must be handled by that single controller. Host port rollup to controller level clearly shows balance between controllers, and virtual disk rollup per controller provides back-end information as well.

Disk group

The most common metric for disk groups is the workload rates averaged over all disks in the disk group, which enables you to quickly determine how heavily loaded the disk group is. As is the case for all metrics, if the time series chart is created, you can monitor the dynamic attributes of the disk group load.

The workload skew across drives within a disk group is good information to have under the heterogeneous disk group conditions as described on page 32. It is not necessarily a problem except under extreme cases of skew and workload. In that case, some definite compromises on array performance might exist.

Performance base lining

The capability to capture, process, and preserve workload and performance metrics for a storage array can be invaluable when encountering problems later or when modifications are considered. This process requires running HP Command View EVAPerf at regular intervals determined by the user and archiving the data. The essence of this practice is to capture all performance logs of interest across the time intervals of interest for an uninterrupted period. The benefits of this practice are many, a few of which are mentioned here.

Normal operation

- Provide history workload and performance characteristics of the target system. This establishes a baseline of the system while it is healthy and therefore provides a reference for when it is not. If periodic (daily, for example) review of the information will be done, then immediate post processing of the data would be required. If only archival information is required in the event of a problem, then the data can be stored in raw form.
- Periodic patterns can be determined from analysis of long-term data. Establishing a daily pattern is essential for characterizing the workload trends and variation throughout the day. Weekly patterns also help to distinguish abnormal behavior from normal variations. Even monthly patterns are useful in some cases.
- Long-term trends can be identified as relevant factors change, some of which are storage occupancy (more data on the subsystem), workload growth (more users on the e-mail system), and environment variables (more incident traffic on shared ISL).
- Current array performance capacity assessment is essential when determining the need and character of potential upgrades. For example, additional physical drives will not help when the dominant workload is large transfer sequential I/O.

When something goes wrong

- Quantify the performance change when a problem is observed by comparing current state against previous activity. Determining if there is an associated workload change at the same instant as the performance metric change is invaluable.
- If the problem is intermittent or periodic, historical sampling enables you to determine precisely when the problem happens and when it does not. This facilitates correlation with other events.
- If performance degrades, the data enables you to determine if there is an associated change in workload. If latencies increase, does the request rate increase or decrease? The introduction of a sequential workload on one controller can affect request latencies on the other.
- Reference data enables you to identify the moment when a problem began to appear. You should be aware that, if correlation is to be made with other reporting mechanisms (for example, Command View event logs, iostat, or Perfmon server metrics), the time references for all host systems must be close, if not synchronized.

Comparison techniques

- Process control charts are commonly used in a manufacturing environment to track some specific set of parameters. This technique employs means and standard deviations to determine if a process is within acceptable limits and graphs easily expose trends.
- With or without process control charts, sometimes you might want to set thresholds for visual or automated detection of a performance issue. Creating upper thresholds for request rates or data rates is problematic because higher is usually better. Lower limits are difficult because most workloads become inactive part of the time or experience periods of less than normal demand or have high variability. Queue depths might realistically employ an upper limit if the workload is well documented and reasonably predictable. Latencies tend to be the most reliable predictor for many workloads because lower is always better and there are common guidelines from many applications for that parameter.

5983-1674EN, 05/2005