HP Storage Essentials

Software Version: 9.4.1

Performance Metrics Guide

Document Release Date: Friday, July 08, 2011

Software Release Date: July 2011

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

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

Copyright Notices

Trademark Notices

Java is a registered trademark of Oracle and/or its affiliates.

Microsoft® and Windows® are registered trademarks of Microsoft Corporation.

Oracle is a registered trademark of Oracle Corporation.

UNIX® is a registered trademark of the Open Group.

Acknowledgements

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

This product includes software developed by the JDOM Project (http://www.jdom.org/).

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).

This product includes software written by Tim Hudson (tjh@cryptsoft.com).

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Or click the **New users – please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at:

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

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

Contents

1 HP Storage EssentialsPerformance Manager

HP Storage Essentials provides the key monitoring and analytical solutions that are essential to administration of critical business applications and the complex storage infrastructures supporting them. With the Performance Manager component, you have immediate visibility into the applications, hosts, switches, arrays, and path structures in your environment.

HP Storage Essentials also provides end-to-end storage area network (SAN) monitoring and reporting capabilities in a rich web-based user interface that allows you to select and display specific path performance metrics. You can view the aggregated results in easy-to-read charts and graphs, in a number of standard out-of-box reports (available using the SRM Report Optimizer), and in ad-hoc custom reports that you can develop with an optional user license. This resulting data gives you the detailed information you need for swift decision-making and efficient resolution of performance issues. As a result, you can increase system administration reaction times, troubleshoot performance bottlenecks faster, and quickly visualize the big performance picture of your array storage, hosts, applications, and SAN infrastructure.

The Performance Manager component provides a single, unified interface that enables you to view performance data appropriate for basic monitoring operations. The built-in capabilities include performance collection against managed databases, managed hosts, VMware ESX, NAS, and fabrics.

For comprehensive performance monitoring of disk arrays, HP recommends optional add-on Performance Packs—the HP Storage Essentials SRM Performance Pack for HP EVA focused on HP StorageWorks Enterprise Virtual Arrays (EVA) and the HP Storage Essentials Performance Pack Enterprise software for HP StorageWorks XP, Hitachi Data Systems (HDS), and EMC Symmetrix/DMX performance management.

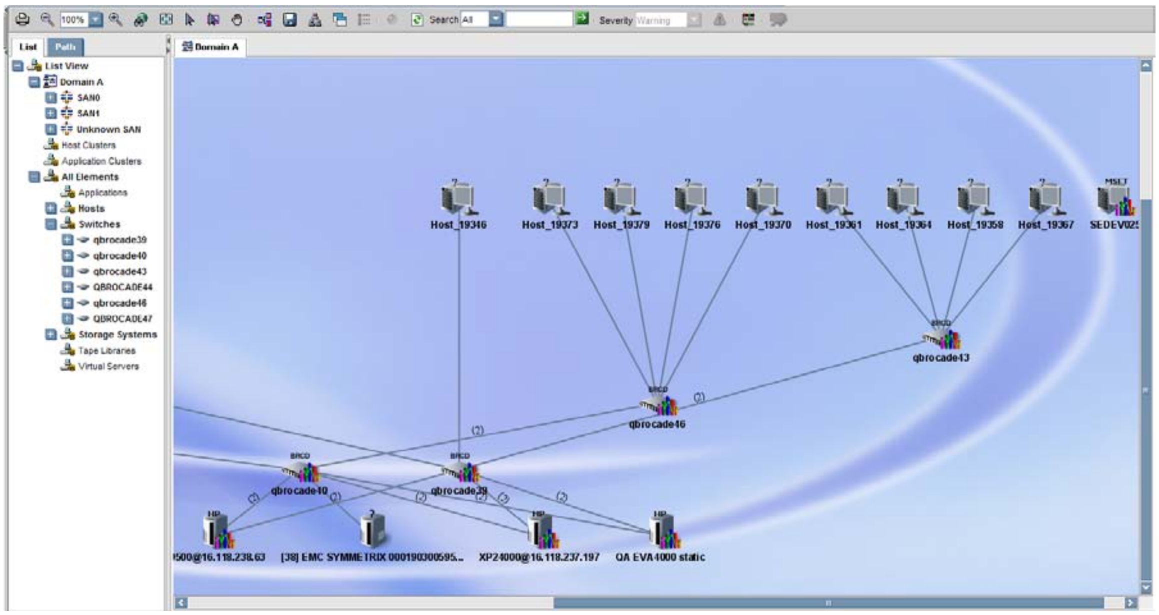
There is also an HP Storage Essentials Performance Pack Enterprise for NetApp. This Performance Pack is installed when you install the NetApp NAS Manager and does not require that you purchase a special license.

To learn more about how Performance Manager supports the day-to-day critical activities of monitoring storage systems, see the following topics about Performance Manager features:

- End-to-End SAN Path Topology on next page
- Flexible Access to Performance Data on page 13
- Path Filtering for Performance Analysis on page 13
- Performance Alerts and Notifications on page 14

End-to-End SAN Path Topology

Performance Manager maps the complete SAN path from business applications through the underlying storage fabric and SAN components to storage devices. From the host application, you can probe through the host bus adapter (HBA) into the switch and finally down to the disk to view all performance characteristics in a single screen.



You can view the performance details for each individual component or aggregate collected data for system-level analytics. Performance diagnostics are available for these path components:

- Business applications
- Host servers
- Host bus adapters
- Fabric switches
- Storage devices (SAN, DAS, and NAS)
- Storage arrays (EVA, XP, HDS, NetApp, and EMC Symmetrix)

Information about each of these performance areas is available in this guide. This guide focuses on performance metrics for disk arrays supported by the HP Storage Essentials SRM Performance Pack and Performance Pack Enterprise plug-ins for the HP Enterprise Virtual Array (EVA), HP XP, Hitachi Data Systems (HDS), and EMC Symmetrix arrays, and NetApp NAS devices.

Performance metrics are automatically collected from systems in the environment that have the appropriate CIM Extension or meet the pre-requisite for performance data collection. See the *HP Storage Essentials Installation Guide* and *User Guide* for data collection requirements.

All devices that meet the pre-requisite for performance data collection display in the HP Storage Essentials Performance Manager navigation map. You can click on any device in the map to display its performance history, or view devices in real time by selecting metrics from systems along the data path. You can also save the user-defined chart workspace and recall it at a later time for analysis of seasonal or reoccurring performance issues for a select group of systems.

HP recommends that you disable collection of unnecessary metrics on a per device basis, and as a best practice, use the sub-15 minute collection feature sparingly. Although you can set individual collectors to as little as 1 minute, these should be set only for limited duration and/or the scope of the collection.

The HP Storage Essentials SRM Performance Pack plug-in is required for performance collection of HP EVA arrays with HP Storage Essentials. The HP Storage Essentials Performance Pack Enterprise plug-in is required for performance management of HP XP or HDS arrays. For more information about HP Storage Essentials requirements for performance collection, see the *HP Storage Essentials Installation Guide* and *User Guide*.

Flexible Access to Performance Data

In addition to monitoring the statistics for each of these resources, HP Storage Essentials gives you three ways to analyze performance data using the Performance Manager:

- Real-time collection – displays tiled graphs that chart the statistics of each performance metric you select and shows you where performance is impacted along the application-storage path with a fine granularity of data points.
- Historical trend – trends performance over time using charts with customizable start and end dates, and hourly, daily, weekly, and monthly snapshots that enable you to quickly see what has changed over time.
- Trend extrapolation – leverages the trend period you selected to forecast what performance will look like in the future if the trend continues.

You are able to view performance data in real time at a fixed 15-, 20-, or 30-second interval by interactively working within the Performance Manager graphical user interface to select individual metrics in the performance charts. Performance data is otherwise collected and displayed at a default 15- or 60-minute interval and aggregated according to the weighted average for the sampling period. To customize a data sampling, you can configure the collection frequency for monitored components as needed.

You will want to take advantage of this flexible access to performance data as you complete your daily routines of verification and reporting, and as you troubleshoot performance issues.

Path Filtering for Performance Analysis

To further accelerate analysis of performance issues, Performance Manager automatically filters out any elements in the storage environment that are not relevant to the path you are monitoring. When you select an application or database instance, you see only the hosts, HBA ports, switch ports, switches, storage system ports, and storage systems, with their respective metrics, that are significant to your inquiry.

Performance Alerts and Notifications

You can send notification e-mails, generate events or run custom command scripts when performance metrics reach thresholds or attain other conditions you've set in a performance policy. This is another way to set up HP Storage Essentials performance metrics to help you monitor performance in your storage environment. For example, if you are monitoring link failures on switches, you can specify in a performance policy that you want to be notified whenever the number of link failures for a specific switch exceeds your specific requirement.

Performance policies and alerts are managed by the HP Storage EssentialsPolicy Manager. By setting policies, you can determine the point at which the data indicates a potential performance issue and be able to respond immediately. For more information about the Policy Manager and performance policy setup, see the *HP Storage Essentials Storage Performance Management Guide*.

Performance Metrics Overview

Performance *metrics*, sometimes called *counters*, are meaningful only within the context of the specific storage element you monitor. For example, if you are collecting performance data for a host bus adapter (HBA), you would be interested in the metric that gives you the number of bytes received and transmitted. But if you are looking at server performance, you would be interested in the CPU utilization and free physical memory.

HP Storage Essentials provides a full range of metrics that you can use to monitor the performance of storage applications and devices in your SAN. These are presented in this guide by specific storage element type and branded vendor.

When you monitor storage elements in Performance Manager, you can view the list of metrics available for each element you select in your SAN topology map. The performance metrics for each SAN element type are described in the following sections of this guide and include metrics for:

- Business applications
- Host servers
- Host bus adapters
- Fabric switches
- Storage arrays

For information about performance management and metrics, see Managing Storage System Performance on page 16.

Performance Data Collection

HP Storage Essentials uses performance collectors to gather information for Provisioning Manager monitoring and charts. From the Data Collection tab you can view all collectors used to gather data for each element. For information about configuring performance collectors, see the *HP Storage Essentials User Guide*.

You can decide the type of data you want collected for each element and the interval (in minutes) between collector runs. You can also start or stop collections by starting or stopping the collector schedule. Although you can set collection intervals to 60 minutes, 15 minutes, or even 1 minute, depending on the level of performance data you require, if you set an interval for performance collection to less than 15 minutes, the system memory usage typically increases and the size of the database on the management server expands.

For this reason, HP recommends that you do not set collection intervals below 5 minutes as a best practice. In those situations where you must do this, limit the scope of the collection by enabling sub 5 minute collection on a limited set of collectors for short durations (hours/day). If you need even finer data granularity for diagnosing, for example a transient condition of limited time span, use the interactive real-time data collection feature available from the Performance Manager graphical user interface.

When determining collection schedules, please be aware that setting collection intervals at less than 5 minutes may have an adverse impact to the device you are monitoring. In other words, if the device is showing poor performance levels before you begin the collection, it is most likely that the resulting impact will show up as dropped performance data points (or gaps in collection), If the collection is performed across a large number of systems concurrently, there is also the possibility of impacting the HP Storage Essentials management server or network.

Aggregation of Performance Data

Performance data collected in real time is immediately visible in charts from the Performance Manager interface. Raw performance data is stored in the database for later viewing up until the preservation interval you've specified. at which point it is rolled up and averaged into hourly data points. All other performance data is collected at 15 or 60 minute intervals by default.

Raw performance data is aggregated over time using the HP Storage Essentials daily collection and roll-up processes to save disk space on the management server..After one day, the data is aggregated to one hour intervals. After a week, it is aggregated to one day intervals, and so forth, up to monthly and yearly aggregations. Values are aggregated according to the weighted average for the sampling period, and a storage administrator can configure the collection frequency for the SAN components within the monitored system.

Customizing Performance Data Collection

Follow these best practices to ensure that you are getting optimum performance data with the least impact on your system's performance.

- Disable collection of metrics on systems that are not business critical.
- Set longer collection times for less important systems (for example, once a day, or 4 times/day).
- Disable collection (where applicable) of specific metrics which are not useful for your reporting or corporate policies.
- Leave the default of 60- or 15-minute collection intervals (where applicable).
- Set collection intervals to as low as 5 minutes for key systems only.

When you use the real-time collection feature of Performance Manager to detail a specific frequency of collection (which is fixed and not user-selectable), follow these guidelines based on device class:

Element Type	Frequency
Application, Host, or Switch	15, 20, or 30 seconds
Storage arrays (with Performance Pack plug-ins)	20 seconds

Managing Storage System Performance

To ensure availability and robust performance of business applications, storage administrators must routinely monitor and manage the arrays and storage area network (SAN) (based on the array) that support those applications. This begins with proactive planning and discovery designed to anticipate problems and enable quick reaction to application or business service slow down. Performance Manager supports this activity by giving an administrator visibility at several points along the data path: applications, host servers, switches, arrays, and array elements such as volumes, disks, controllers, and storage subsystem Performance Manager metrics for each of these path points enable you to collect the statistics, perform trending, and issue reports that support the complex decision-making process that gets the performance results your company needs.

Performance Management and Metrics

HP recommends the following performance management methodology as an approach to utilizing Performance Managermetrics to their greatest advantage when you find a performance issue. This section describes this approach using these four easy steps:

Step 1: Identify the Physical Path

Identify the physical path along which the performance issue exists. HP Storage Essentials discovers the storage environment and helps visualize the topology and paths in the storage environment. It depicts the path from application—to server—to virtual elements—to volumes—to HBA—to switches—to array. This automatic filtering helps you identify quickly the systems along a path where the performance problem exists.

Step 2: Isolate the Bottleneck

Isolate the elements on the path that cause the performance bottleneck. With the path along which the problem exists displayed, you can drill down to view the details of every port and infrastructure element involved in the data path and isolate the element that is the performance bottleneck. The isolated element can now undergo intense troubleshooting based on deep inquiry using performance metrics.

Step 3: Troubleshoot the Cause

Troubleshoot the element to understand the root cause. This guide details the important metrics to look at for each isolated element in the path. For example, if you suspect that a host server's workload is too high, and the CPU is underpowered for the load, refer to the Host Metrics section to determine which metrics will give you the information you need to confirm this.

Use metrics also to proactively measure and present indisputable data related to storage performance that gives you the ability to establish baselines and compare over time such statistics as percentage of usage, I/O throughput, available storage space, and workload trends.

Step 4: Report on Performance Trends and Statistics

The Performance Manager GUI provides standard reporting capabilities (including trending) that allow you to view application, host, switch and array metrics. Look at trends and generate reports as a pre-step to making informed decisions. Use history-based trends and dynamic reporting to collect performance data and build evidence which points to sustainable solutions. A set of out-of-box performance reports can be leveraged for decision-making purposes and future planning.

You can also use the HP Storage Essentials Report Optimizer for viewing host, switch, and array metrics. Report Optimizer user licenses can be purchased for ad-hoc reporting against metrics exposed in those cases where out-of-box reports are not sufficient. For a list of performance reports in the Report Optimizer Report Pack, see the *Report Optimizer Quick Start Guide*.

Step 5: Configure Proactive Threshold Notification using Policy Manager

After you establish a baseline, leverage the HP Storage Essentials Policy Manager by setting up performance policies which provide the early warnings needed to address future performance issues before they happen. The Policy Manager provides policy templates for many of the collected performance metrics. Use performance policies to define thresholds and resulting actions (such as generating events, emails or running scripts) when any given threshold is exceeded. For information about using the HP Storage Essentials Policy Manager and setting up performance policies, see the *HP Storage Essentials Storage Performance Management Guide*.

Ways to Use This Guide

This guide describes the performance management metrics collected, trended, and reported by the HP Storage EssentialsPerformance Manager. It contains important recommendations and best practices for monitoring the performance of your storage systems, and includes metric reference tables for each storage element in your path topology. This information can be useful when interpreting metrics data, analyzing performance states, and identifying performance issues for each of your SAN components.

Use the introductory sections to learn about what to look for when troubleshooting performance problems, and which metrics to monitor closely as indicators of potential problems. Use the metrics reference tables to understand which metrics are available for which storage element, and note any special considerations or conditions for their usage.

Additional Resources

For more information about the HP Storage Essentials, see the *HP Storage EssentialsStorage Performance Management Guide, Installation Guide*, and *User Guide* located at:

http://h20230.www2.hp.com/selfsolve/manuals

You will find HP whitepapers related to performance management and monitoring at:

http://h18006.www1.hp.com/storage/softwhitepapers.html

For additional HP Enterprise Virtual Array (EVA) performance information, see documents located at:

http://h71028.www7.hp.com/ERC/downloads/5983-1674EN.pdf

http://kb.know.hp.com/lib/KB04000/KB3903.docx

The HP StorageWorks Sizer Tool is located at:

http://h30144.www3.hp.com/SWDSizerWeb/default.htm

To verify compatibility of component and software versions, see the HP Storage Essentials support matrix document on your product DVD. You can also find the support matrix document in the Documentation Center (**Help** > **Documentation Center**) accessible from the HP Storage Essentials user interface.

2 Application Performance Metrics

This section describes the HP Storage Essentials performance management metrics for the applications running in your storage environment. These performance metrics are gathered by data collectors and available for reporting. It also includes techniques and best practices for managing performance of your applications.

HP Storage Essentials provides metrics for the following applications:

- Microsoft Exchange Server Performance Metrics on page 21
- Oracle Server Performance Metrics on page 25
- Microsoft SQL Server Performance Metrics on page 27
- Sybase Performance Metrics on page 29
- IBM DB2 Performance Metrics on page 31
- Informix Performance Metrics on page 30
- InterSystems Caché Database Performance Metrics on page 32

Troubleshooting Application Performance

Follow these steps to monitor and/or troubleshoot host application performance using the HP Storage EssentialsPerformance Manager. The instructions assume that you launched the Performance Manager and have a displayed view of the storage path topology for your system. To begin, navigate to the host server that runs your application.

Step	Description	Considerations
1	Check the host processor utilization and physical/virtual memory consumption by selecting to view these metrics: • Free Physical Memory (KBytes) • Free Virtual Memory (KBytes) • Processor Utilization (%)	If your processor is running near 100 percent and virtual memory consumption is greater than physical memory, the problem might be that the host does not have sufficient resources to process the I/O for the application.
2	Select the application path where the application is running, and expand its members.	

Step	Description	Considerations
3	Check the logical device (LDEV) read/write response time. Note if the LDEV has an extremely high response time (above 20 ms) or zero response.	If the response time is higher than expected, this can indicate a resource contention problem within the array. If there is zero response, check for a link failure.
4	Chart the adapter, switch, and array port Bytes Received and Bytes Transmitted metrics on the same graph to verify that traffic is passing through to the array without problems.	If you see large differences between the host bus adapter (HBA), switch and array port, the problem could be a bad SFP or fiber link.
5	Chart the array port and LDEV Total Data Rate.	The array port should have at least the same data rate as the LDEV. If the array port data rate is lower than LDEV that could indicate you have too much write pending in cache.
6	Select the array and chart its cache write pending percentage.	The array will begin to hold back I/O If the write pending is at or above 70 percent. Some of the causes for excessive write pending could be too much I/O from other devices in the Cache Logical Partition (CLPR), replication processing, and external storage (unless you have cache write-through mode enabled).

Improving Microsoft Exchange Server Performance

There are multiple methods for improving Microsoft Exchange Server performance. Techniques range from placing databases and transaction logs in an optimal location to using the performance metrics in this document to locate and remove Microsoft Windows bottlenecks.

Refer to the Microsoft TechNet library (http://technet.microsoft.com/en-us/library/default.aspx) for information about tuning your Microsoft Exchange Server. Use the information in the following sections of this guide to learn more about using HP Storage Essentials performance metrics to identify areas for improvement.

Performance Baseline

As a part of your initial data collection, create a baseline for your Microsoft Exchange servers. This baseline assumes that there are no serious problems with the servers when you take the readings. The baseline values tell you what is normal for your Microsoft Exchange servers and provide reliable values against which you can compare future measurements.

Baseline values are not valid forever however. Any time you perform a major action on your Microsoft Exchange server—such as installing a service pack or adding new hardware—the next reading should be compared to the original baseline value to determine the impact of the change. After this comparison is made, the most recent reading then becomes the new baseline.

Even if you do not make any major changes to the Microsoft Exchange server, HP recommends that you update your baseline values every four to six months by making your most recent reading the new baseline. In this way, you always have a semi-current (and realistic) baseline from which to work.

Activity Spikes

Spikes in activity are normal, even for a well-functioning server. If you are only collecting data summaries every 15 minutes, it is possible that some samples will occur during big spikes of activity, and make the Microsoft Exchange server look busier than it actually is.

To get a more accurate reading of the data, take a data collection 15 minutes and then again before 15 minutes after an activity spike. Look at those performance metrics which give you average measurements for the data over that specific time period. Not that this does not show you the spikes/transients, because these will have been averaged into the final measurement. If you want to collect more detailed statistics at a finer level of granularity, collecting data down to 5-minute intervals will better reflect actual performance during a given period. You can also collect data in real time for a typical 20-second collection by working interactively with the HP Storage Essentials interface.

When you look at the collected data for a week and if the averages seem high, it is possible that the data was collected during periods of high activity. Common advice is not to worry about it. The purpose of data collection is to look for long-term trends so that you can forecast your future hardware needs.

HP Storage Essentials provides roll up of data from raw (sub-hourly) data, to hourly, daily, weekly, and so forth up to yearly aggregation of data. Because the raw (sub hourly) is only preserved for one day before being rolled up and summarized, any spikes (where they were apparent during smaller interval sampling) would become less apparent and averaged into the larger pool of data with each roll-up. If weekly views of your server performance is a critical piece of your reporting, you can set preservation of RAW to two weeks, and set the collection interval to, for example, 5 minutes, and get a detailed view of the previous week for the elements selected.

HP recommends that you do not set RAW preservation longer than the defaults listed in the GUI (1 day). If you need to adjust the default intervals, do so only with the assistance of HP Support.

Microsoft Exchange Server Performance Metrics

This table lists the metrics used by HP Storage Essentials to measure performance for the Microsoft Exchange Server. An asterisk (*) marks a metric that does not collect historical data. All other metrics use collected historical data to provide statistics.

Metric	Description	Units	Use to...
Application Server			
Exchange Services	Services Running/ Not Running (Site Replication, Routing Engine, POP3, Exchange Management, MTA Stacks, Info Store, IMAP4, Events)	Count	Show whether or not a critical Microsoft Exchange service is running.
Final Destination currently unreachable Queue Size*	Contains the number of messages that cannot reach their intended destination.	Count	Set alerts. Possible reasons why final destination is not reachable are: • No route • Microsoft Exchange connector unreachable • No remote delivery queue available
Messages awaiting directory lookup Queue Size*	Messages awaiting directory lookup queue size. These are messages sent to a Microsoft Exchange 2000 or 2003 server.	Count	Set alerts or troubleshoot Event ID 9035, 6004, 9003, 9004. Possible causes: Message Archiving, Insufficient Permissions, or global catalog server issues.
Messages to be routed Queue Size*	Number of messages to be routed	Count	Set alerts or troubleshoot message routing issues in Microsoft Exchange.
Pre-submission Queue Size*	Pre-submission queue size	Count	Measure the number of messages waiting in the pre-submission queue. Possible causes are: • Third-party applications • SMTP events • DNS
SMTP Local Delivery Queue Size*	Number of messages in the local queue	Count	Set alerts. Large numbers of messages indicate possible disk I/O issues or dismounted stores.

Metric	Description	Units	Use to...
SMTP Server Queue Summary*	Summary of local and remote queues	Count	Set alerts and notifications. Helps pinpoint disk, DNS, or network I/O issues.
Storage Groups			
Log Size (MBytes)	Storage group log file size	MB	Help manage log file sizes in Microsoft Exchange.
Storage Group Size	Size of the Storage Group in MB.Contains one or more mailbox or public folders, transaction logs, and system files depending on the version.	MB	Provide a facility to track the Microsoft Exchange group utilization from a higher level than only mailbox or folders sizes.
Message Stores and Public Folders			
*Active Client Logins	Number of logons active (issued by any MAPI requests) within a specific time interval	Count	Track the number of sessions/users accessing the Microsoft Exchange server at a given time.
*Average Delivery Time	Rate at which messages are delivered to all recipients	Msg/s	Measure long delivery times that could be caused by deferred delivery queues or stalled messages.
Mail Related Objects Count	Public folder mail related objects count.	Count	Count the number of mail messages, appointments, meeting requests, tasks, task requests, contacts, remote mail items, and notes.
*Receive Queue Size	Number of messages in the mailbox store's receive queue	Count	Track MTA queues, IS queues, and SMTP queues. Possible issues are: • Related service • Looping or corrupt message • Source server issue

Metric	Description	Units	Use to...
*Send Queue Size1	Number of deferred delivery or submission messages.	Count	Track MTA queues, IS queues, and SMTP queues. Possible issues are: • Related service • Looping or corrupt message • Source server issue
Store Size	Mailbox store size	MB	Set alerts. Helps alert administrators when a store size is reaching its limits.

*Not supported with Microsoft 2007.

Monitoring Oracle Server Performance

You can use Oracle application performance metrics in HP Storage Essentials in association with other storage-related performance metrics to identify potential bottlenecks. The metrics available in HP Storage Essentials have been carefully chosen to provide this capability and are not a replacement for native tools.

Consider the following recommended steps when tuning Oracle applications for improved performance. To prevent unanticipated tuning side effects, please maintain the order in which steps are listed in this section. For example, it is not a good practice to increase the buffer cache if you can reduce I/O by rewriting a SQL statement.

Step 1. Review the Database Design

A well-designed database schema is the first step to solid application performance. If you are seeing poor system performance, this is usually the result of poor database design. If if it still possible, you might consider database design improvements that include the following:

- You should generally normalize to the 3NF. Selective de-normalization can provide valuable performance improvements.
- Always keep the "data access path" in mind when you are designing.
- Look at proper data partitioning, data replication, and aggregation tables for decision support systems.

Step 2: Tune the Application

Experience shows that approximately 80% of all Oracle system performance problems are resolved by coding optimal SQL. You can also enhance performance by scheduling batch tasks and other large workloads after peak working hours.

Step 3: Tune the Memory

Make sure to properly size your database buffers (for example, shared_pool, buffer cache, log buffer) by looking at performance metrics related to wait events, buffer hit ratios, system swapping, paging, and so forth. You might also want to pin large objects into memory to prevent frequent reloads.

Step 4: Tune the Disk I/O

To provide maximum disk subsystem throughput, properly size and place database files. Also review performance metrics for frequent disk sorts, full table scans, missing indexes, row chaining, data fragmentation, and so forth.

Step 5: Eliminate Database Contention

Application contention for database resources can slow application response time significantly. Carefully review database locks, latches, and wait events and eliminate them wherever possible.

Step 6: Tune the Operating System

Monitor and tune the operating system, CPU, and I/O and memory utilization. For information about best practices for doing this, read the related Oracle FAQ related to your specific operating system at http://www.orafaq.com/wiki/Operating_system.

Oracle Server Performance Metrics

The following table lists the HP Storage EssentialsPerformance Manager metrics available for monitoring Oracle performance and describes what they count as well as considerations for using them.

Metric	Description	Units	Use to...
Buffer Hit Ratio	Percentage (%) of requests against data block buffer	%	Measure the effectiveness of the Oracle data block buffer. Used with a database that has an undersized DB Cache size where the “working set” of frequently-referenced data has not been cached.
Dictionary Hit Ratio	Ratio of logical reads to physical disk reads	%	Monitor ratio balance between logical and physical disk reads. As the hit ratio approaches 100 percent, more data blocks are found in memory, resulting in fewer disk I/Os and faster overall database performance.
File Read Percent	Percentage (%) of sequential reads.	%	Measure the performance and efficiency of indexes.

Metric	Description	Units	Use to...
File Total I/O Percent	Percentage (%) of sequential reads and writes	%	Capture overall sequential database I/Os.
File Write Percent	Percentage (%) of sequential writes	%	Identify other bottlenecks like redo waits, bind variables, bulk operations, and index contention.
In Memory Sort Ratio	Percentage (%) of sorts (from ORDER by clauses or index building) that are done to disk compared to in-memory	%	Differentiate and identify the number of disk sorts performed in TEMP tablespace versus those performed in-memory (RAM sorts).
Library Cache Hit Ratio (%)	Monitors the percentage (%) of entries in the library cache that were parsed more than once (reloads) over the lifetime of the instance.	%	Set the shared_pool_size large enough to prevent excessive re-parsing of SQL.
Parse CPU to Total CPU Ratio	Percentage (%) of reentrant SQL statements compared to re-parsed SQL statements; also referred to as the Parse to Execute ratio	%	Show the number of unique incoming SQL statements or that SQL statements are NOT reentrant.
Redo Buffer Allocation Retries	Total number of retries needed to allocate space in the redo buffer	%	Monitor the buffer allocation retries over a period of time while the application is running. If the Redo Buffer Allocation Retries value is continuously increasing, then increase the LOG_ BUFFER value.
System Event Time Waited	Provides wait state details for ongoing Oracle transactions.	ms	Analyze time-based events, and system-wide and session wait events.
Tablespace Read Percent	Percentage (%) of tablespace reads over an interval	%	Monitor tablespace usage and growth.
Tablespace Total I/O Percent	Percentage (%) of total tablespace I/O over an interval	%	Monitor tablespace usage and growth.

Metric	Description	Units	Use to...
Tablespace Write Percent	Percentage (%) of tablespace writes over an interval	%	Monitor tablespace usage and growth.

Monitoring Microsoft SQL Server Performance

Microsoft provides information about tuning Microsoft SQL Servers for improved performance in the Microsoft TechNet library (http://technet.microsoft.com/en-us/library/default.aspx). Use this guide to understand how to use Storage Essential performance metrics to help you identify bottlenecks and spot potential problem areas before they affect your system.

Using Metrics to Identify Bottlenecks

Routine monitoring of your Microsoft SQL Server applications for performance bottlenecks is important. One bottleneck can manifest itself as other bottlenecks, and after you remove one, you might find the SQL Server is constrained by a completely different bottleneck. For example, on an SQL Server with disk constraints (a low RPM drive) and a small amount of available memory (causing excessive paging), you will find high processor utilization because the SQL Server processor is spending a large amount of time paging to disk. You can consider adding more memory to reduce the amount of paging, but then the disk bottleneck might also become the primary constraint.

Disk bottlenecks tend to manifest themselves in high processor utilization situations because the processor is busy managing disk I/O. Memory bottlenecks manifest themselves as high processor and disk utilization. This results because the processor is busy managing virtual memory and disk I/O. As a best practice, when examining bottlenecks, begin by examining memory bottlenecks, then disk bottlenecks and finally processor bottlenecks.

Microsoft SQL Server Performance Metrics

This table describes the performance metrics available for Microsoft SQL Server 2005 and 2008. For a list of supported software versions for Microsoft SQL Server, see the HP Storage Essentials support matrix document located in the Documentation Center (**Help** > **Documentation Center**).

Metric	Description	Units	Use to...
Blocked Processes	Number of blocked processes in your Microsoft SQL Server database	Count	Flag the database administrator (DBA) that a number of blocked processes are present. The DBA can then use a SELECT statement to identify the offending processes.

Metric	Description	Units	Use to...
Buffer Cache Hit Rate	Percentage (%) of pages found in the buffer cache. The % is calculated as the total number of successful cache hits divided by the total number of requested cache lookups.	%	Monitor the buffer cache hit ratio. After a system maintains a steady state of operation, this metric should achieve rates of 90 percent or higher. The buffer cache hit ratio can be increased by increasing the amount of memory available to SQL Server.
CPU Usage Percentage	Percentage (%) of the total available CPU time that Microsoft SQL Server uses during the current interval.	%	Identify potential performance issues. A high CPU usage rate can indicate performance bottlenecks. Options for resolving the bottleneck are to add multiple CPUs, allocate resources more efficiently, identify resource intensive applications, and reduce workloads, or upgrade CPUs.
Cache Memory	Total amount of dynamic memory the server is using for the dynamic SQL cache.	KB	Determine if the current cache memory is meeting the needs of the SQL Server application.
Dead Locks	Number of lock requests per second that resulted in a deadlock. A deadlock is a situation where two or more competing actions are waiting for the other to finish.	Req/s	Generate an alert for a deadlock condition. You can then identify the SPIDs and the resources that are involved in a deadlock using the database logs.
Lock Requests	Number of new locks and lock conversions per second requested from the lock manager.	Req/s	Identify issues related to data retrieval. A high number of lock requests with low request rate is an indicator that SQL Server must do table scans when retrieving data.
Lock Time Outs	Number of lock requests per second that timed out, including internal requests for NOWAIT locks.	Req/s	Identify areas of lock contention and congestion in the database.

Metric	Description	Units	Use to...
Lock Waits	Lock wait time is the time that a process spends waiting for another process to release a lock.	Wait/s	Generate an alert to the database administrator when the number exceeds pre-set thresholds; also indicates database congestion.
Memory Usage Percentage	Number of pages in the procedure cache that are currently allocated to a process.	%	Monitor allocation of memory to processes. A negative number indicates that a process is freeing memory allocated by another process.
Physical I/O Percentage	Percentage of elapsed time that the disk drive was busy servicing read or write requests	%	Identify disk I/O problems in the SQL server instance. A value greater than 50 percent may indicate an I/O bottleneck.
Plan Cache Hits Ratio	Ratio between plan cache hits and lookups. The plan contains stored procedures, ad hoc and prepared Transact-SQL statements, and triggers.	%	Monitor the database plan cache performance.
Target Server Memory	Amount of dynamic memory the server can consume	KB	Manage database server memory.
Total Server Memory	Committed memory from the buffer pool. This is NOT the total memory used by the SQL Server.	KB	Manage database server memory.
Transactions	Total number of database transactions	Count	Monitor and manage load balancing.
User Connections	Total number of database user connections	Count	Identify the number of users who are accessing the database.

Sybase Performance Metrics

This table lists the performance metrics available for Sybase Adaptive Server Enterprise systems.

Metric	Description	Units	Use to...
CPU Usage Percentage	Percentage of the total available CPU time that Sybase instance uses during the current interval.	%	Monitor the CPU usage of the Sybase instance. A high CPU usage rate can indicate performance bottlenecks. If you identify an issue, some options are to add multiple CPUs, allocate resources more efficiently, identify resource intensive applications, reduce workload, or upgrade CPUs.
Memory Usage Percentage(%)	Number of pages in the procedure cache that are currently allocated to a process	%	Monitor memory allocation to active processes. A negative number indicates that the process is freeing memory allocated by another process.
Physical I/O Percentage	Percentage of elapsed time that the disk drive was busy servicing read or write requests.	%	Identify disk I/O problems in the Sybase instance. A value greater than 50 percent may indicate an I/O bottleneck.

Informix Performance Metrics

The Read Cached (%) metric is an important database performance indicator. As the hit ratio approaches 100%, this indicates that more data blocks are being found in memory, which in turn results in fewer disk I/Os and faster overall database performance. Use Chunk Read and Chunk Write to monitor chunk usage, and DBSpace Reads and DBSpace Writes to monitor I/O requests to the database.

The following table describes the performance metrics for Informix.

Metric	Description	Units
Buffer Read	Number of reads from the buffer cache by the database server	count
Buffer Write	Number of writes to the buffer cache by the database server.	count
Chunk Read	Total number of reads from the chunk. Use in combination with Chunk Write metric to monitor chunk usage.	count
Chunk Write	Total number of writes to the chunk. Use in combination with Chunk Read metric to monitor chunk usage.	count

Metric	Description	Units
DBSpace Reads	Total number of read calls that involve this dbspace. Use in combination with DBSpace Writes metric to monitor dbspace usage.	count
DBSpace Writes	Total number of write calls that involve this dbspace. Use in combination with DBSpace Reads metric to monitor dbspace usage.	count
Disk Reads	Total number of read operations from disk by the database server	count
Disk Writes	Total number of write operations to disk by the database server	count
Page Reads	Number of pages read from disk by the database server	count
Page Writes	Number of pages transferred to disk by the database server	count
Read Cached	Percentage of all read operations that are read from the buffer cache without requiring a disk read by the database server, calculated as follows: 100 x ((buffer_reads — disk_reads) / (buffer_reads)) As the hit ratio approaches 100%, more data blocks are found in memory. This results in fewer disk I/Os and faster overall database performance.	%
Write Cached	Percentage of all write operations that are buffer writes by the database server, calculated as follows: 100 x ((buffer_writes — disk_writes) / (buffer_writes))	%

IBM DB2 Performance Metrics

This table lists the performance metrics tracked for IBM DB2 databases.

Metric	Description	Units	Use to...
Instance Private Sort Memory	Private sort memory is calculated using the formula `(db2.sort_heap_allocated / sheapthres) x 100`. In this instance, sort_heap_allocated is the system monitor element, and sheapthres is a DBM configuration parameter.	%	Track private sort memory utilization. Used to check that there is sufficient heap space to perform sorting and that sorts do not overflow.

Metric	Description	Units	Use to...
Instance Monitor Heap Utilization	Utilization is calculated using the formula `(db2.pool_cur_size / db2.pool_max_size) x 100` for the Memory Pool Identifier SQLM_HEAP_MONITOR.	%	Track the consumption of the monitor heap memory, If this percentage reaches the maximum 100%, monitor operations might fail.
Database Catalog Cache Hit Ratio (%)	Hit ratio indicates (as a percentage %) how well the catalog cache is working to avoid actual accesses to the catalog on disk	%	Monitor the ratio between catalog cache and physical disk reads. As the catalog cache hit ratio approaches 100 percent, more catalog blocks are found in memory, resulting in fewer disk I/Os and faster overall database performance.
Database Lock List Utilization	There is one lock list per database, and it contains the locks held by all applications concurrently connected to the database. The indicator is calculated using the formula `(db.lock_list_in_use / (locklist x 4096)) x 100.`	%	Track the amount of lock list memory that is being used.
Database Deadlocks	Number of lock requests that resulted in a deadlock. A deadlock is a situation where two or more competing actions are waiting for the other to finish.	Lock requests	Track deadlock conditions. The DBA will identify the SPIDs and the resources that are involved in a deadlock using the database logs.
Database Shared Sort Memory Utilization	Shared sort memory is calculated using the formula `(db.sort_shrheap_allocated / sheapthres_shr) x 100.` In this instance, sheapthres_shr is a database configuration parameter.	%	Track shared sort memory utilization for the database. Can be used to determine an appropriate value for the shared sort memory threshold.

InterSystems Caché Database Performance Metrics

The following table lists the performance metrics in HP Storage Essentials for InterSystems Caché databases.

Metric	Description	Units
Application Errors	Number of application errors logged Not supported for Caché 5.0.	Count
Global Kills	Number of global kills since startup Only supported for Caché 5.0.	Count
Cache Efficiency	Most recently measured cache efficiency. This is measured as Global references / (physical reads + writes).	GlobalRef/IO
Global Refs	Number of Global references since system startup.	Count
Global Sets	Number of Global Sets and Kills since system startup	Count
Globals Per Second	Most recently measured number of Global references per second Not supported for Caché 5.0.	GlobalRef/s
Disk Reads	Number of physical block read operations since system startup	Count
Disk Writes	Number of physical block write operations since system startup	Count
Journal Entries	Number of entries written to the system journal	Count
Logical Requests	Number of logical block requests since system startup	Count
Routine Loads	Number of routine loads since system startup Only supported for Caché 5.0.	Count
Routine Refs	Number of routine loads and saves since system startup. Not supported for Caché 5.0.	Count
Routine Saves	Number of routine saves since system startup. Only supported for Caché 5.0.	Count

3 Host Server Performance Metrics

By selecting a SAN host server component, you can view processor and memory utilization, host bus adapter (HBA) status, transmission metrics, and application path metrics. The application path maps from the mount point all the way to the array LDEV. Along the path you can choose any device to view its performance.

HP Storage Essentials provides the following metrics for host server performance management, and these are described in this guide:

- Host Server Performance Metrics on next page
- Host Disk Performance Metrics on page 37
- ESX Server Performance Metrics on page 38

Finding Host Performance Bottlenecks

A performance bottleneck is identified by an impaired or stopped flow of data which is measured, for example, in transaction response times, number of I/O requests, I/O sequences of reads and writes, and so forth. In general, bottlenecks are characterized by poor throughput, high data or resource contention, and/or inefficient handling of heavy workload demands. Bottlenecks can occur within the internal resources of the server itself, at some identifiable point along the communications network to which the server is attached, or within the storage fabric utilized by the server. Finding bottlenecks and resolving the issues that cause them is essential to keeping your servers and systems healthy and continually available to users.

HP Storage Essentials does not automatically identify hotspots or bottlenecks. The HP Storage Essentials metrics, however, can help you find bottlenecks by providing the critical iterative statistics that allow you and your managers to see performance levels on your servers and systems over time. Understanding how to configure performance metrics and interpret the data that HP Storage Essentials provides is a first step. The metrics for each of the most common bottleneck areas are described below.

Restrictions

For the Performance Manager to collect and display performance metrics, at least two data points for the hosts are needed. If only one data point is available, no performance metrics are presented.

Not all metrics are supported on all operating systems or for all storage system elements. Refer to the *HP Storage EssentialsUser Guide* for information about supported operating systems and metrics.

Host metrics for SAN-attached volumes are not available in the Performance Manager. SAN volume metrics are not collected from hosts. For volumes from volume manager groups (logical volumes), metrics are presented for only the first volume on that disk, although it might appear as if the metrics for all of the volumes are presented.

Random Access Memory

Use the Free Physical Memory and Virtual Memory Used metrics to measure performance related to the server random access memory (RAM). The Free Physical Memory counter displays the amount of physical RAM available to the Microsoft Windows operating system. The Virtual Memory Used counter measures "hard" page faults which are the times when data had to be swapped between the hard drive's virtual memory and the physical memory on the host.

If the Free Physical Memory counter drops below 10 percent of the RAM, while the Pages/sec counter increases significantly, this might indicate that the server does not have enough RAM to support the software programs that are currently running on the machine.

Central Processing Unit

Use the Processor Utilization % Time metric to track usage levels of the central processing unit (CPU).

The Processor Utilization % Time counter tells you what percent of CPU capacity is being used at any specific point in time. It is common for software application launches and other events to create utilization spikes between 90 and 100 percent. However, if this counter consistently measures over 80 percent, it is possible that your processor lacks the ability to handle your system workload. Because high Processor Utilization % Time metrics for a server typically result in slow response times and weak performance of applications, you might want to consider upgrading your CPU. See Host Server Performance Metrics below.

Physical Disk

To understand the metrics for physical disk usage, also called direct-attached storage or local disk, by your server, use the Disk Utilization % Time metric.

The Disk Utilization % Time counter displays the amount of time the hard drive spends reading or writing data. A disk that runs 40 to 50 percent of the time might need to be replaced or upgraded. Another solution is to consider adding additional storage disks.

If you have the HP Storage Essentials SRM Performance Pack or the HP Storage Essentials Performance Pack Enterprise software, see the *HP Storage Essentials Storage Performance Management Guide* for information about collecting data for performance metrics. You can gather information from volumes at the array level, but not at the host level. See Restrictions on previous page and Host Disk Performance Metrics on the facing page.

Host Server Performance Metrics

Server performance metrics are collected by the CIM Extension on managed hosts. Performance metrics are not available for inferred or unmanaged servers.

You can use the following metrics to measure host performance.

Metric	Description	Units	Use to...
Free Physical Memory	Amount of physical memory available	KB	Measure available main memory for additional processes and threads.
Free Virtual Memory	Amount of virtual (paged) memory available	KB	Measure physical memory optimization and availability over time.
Physical Memory Used	Percentage (%) of main memory used by processes running on the host	%	Track memory utilization trends.
Processor Utilization	Total CPU utilization percentage (%) for all processes running on the host	%	Identify CPU bottlenecks.
Virtual Memory Used	Percentage (%) of paging file in use	%	Determine if a system has insufficient physical memory.

Server performance metrics might not be available for all supported operating systems.

Host Disk Performance Metrics

You can use the following metrics to measure disk performance on a server.

Metric	Description	Units	Use to...
Disk Read	Average time in seconds to read data from disk	KB/s	Compare read times for a given application (for example, read compared to writes).
Disk Total	Total read and write requests in seconds	KB/s	Test maximum throughput.
Disk Utilization	Based on the IRP (I/O request packets) round trip times the Average Disk Sec/Transfer. Indicates how busy a physical disk is over time.	%	Determine the average disk utilization for a given application or known number of processes. Utilization indicates how busy a disk is.
Disk Write	Average time in seconds to write data to disk	KB/s	Compare write times for a given application (for example, writes compared to reads).

Disk performance metrics might not be available for all supported operating systems.

HBA Performance Metrics

The following metrics measure the performance of host bus adapters (HBA).

Metric	Description	Units	Use to...
Bytes Received	Inbound traffic in megabytes per second at the HBA port.	MB/s	Measure network traffic for load balancing, multi-path optimization, and network performance.
Bytes Transmitted	Outbound traffic in megabytes per second at the HBA port	MB/s	Measure network traffic for load balancing, multi-path optimization and network performance.
CRC Errors	Cyclic Redundancy Check errors over a period	Errors	Isolate Cyclic Redundancy Check (CRC) errors on a specific initiator or between devices.
Link Failures	Link Failures over a period	Failures	Isolate connection failures and the effect on performance.

HBA metrics might not be available for all operating systems or all HBAs.

ESX Server Performance Metrics

HP Storage Essentials tracks the following performance metrix for the ESX Server. These metrics are collected agentlessly; that is, the CIM Extension is not required on the ESX Server to view the HP Storage Essentials performance metrics.

Metric	Description	Units	Use to...
Free Physical Memory	Amount of physical memory available	KB	Measure available main memory for additional processes and threads.
Physical Memory Used	Physical memory being consumed by all processes.	%	Indicate physical memory optimization and availability over a period of time.
Processor Utilization	Total CPU utilization (%) for all processes running on the host	%	Identify CPU bottlenecks.
Disk Read	Average time in seconds to read data from disk	KB/s	Compare read times for Virtual Machines and their applications.
Disk Total	Total read/writes in seconds	KB/s	Test maximum throughput

Metric	Description	Units	Use to...
Disk Write	Average time in seconds to write data to the disk	KB/s	Compare write times for Virtual Machines and their applications.
Bytes Transmitted*	Inbound traffic in megabytes per second through the HBA.	MB/s	Measure network traffic for load balancing, multi-path optimization, and network performance.
Bytes Received*	Outbound traffic in megabytes per second through the HBA	MB/s	Measure network traffic for load balancing, multi-path optimization, and network performance.
CRC Errors*	Cyclic Redundancy Check (CRC) errors over a period of time	Errors	Isolate Cyclic Redundancy Check (CRC) errors on a specific initiator or between devices.
Link Failures*	Link Failures over a period of time	Failures	Isolate connection failures and effect on performance.

*Information displayed for the metric is discovered from the fabric switches, and not from the ESX server.

4 Fabric Switch Performance Metrics

By selecting a SAN fabric switch, you can view the aggregated bytes transmitted and received, and the individual switch port metrics including Cyclical Redundancy Check (CRC) errors and link failures. Additionally, you can navigate to other switches, hosts, and arrays connected to the switch and view their metrics.

See Switch Performance Metrics on page 44 for a list of HP Storage Essentials performance management switch metrics.

Identifying Fibre Channel Performance Issues

Fibre channel performance issues can be identified by performing a Cyclic Redundancy Check (CRC). CRC is a method of data integrity assurance across a transmission link. On the transmitting end, a mathematical computation is performed on the bitstream, and the result is added to the data frame. The process is reversed on the receiving end. If the two results do not match, a CRC error is generated, resulting in retransmission of the frame to maintain data integrity.

FC Errors

The following are examples of Fibre Channel (FC) errors that may be observed as a result of the above issue. This issue may not be the only reason that such errors occur. Other types of Fibre Channel errors could also potentially occur:

- Disconnects to Fibre Channel-attached storage during high I/O traffic.
- During high I/O load, CRC errors occur in conjunction with a Microsoft Windows error message: device not accessible.
- ProLiant BL20p G3 server blades stop responding with "Link failure," "loss of sync" or "loss of signal" errors logged at the switch when the HBA performs a link reset.
- Multiple path failures in multi-path environments.

Note: Because the Performance Manager does not support host events, you will need to use native tools, such as logs, event viewer, and so forth, to see host events.

CRC Errors

Brief Cyclic Redundancy Check (CRC) errors in HP Storage Essentials are a normal occurrence when an HBA is first powered on or off, or when cables are attached or detached. Excessive CRC errors during data transfers can cause performance degradation but do not compromise data integrity.

Link Failure

Link failure is the result of a loss of signal, loss of synchronization, or NOS primitive received. A link failure indicates that a link is actually "broken" for a period of time. It can possibly be due to a faulty connector, media interface adapter (MIA), or cable. The recovery for this type error is disruptive. This error is surfaced to the application using the SAN device that encountered this link failure. This causes the system to run degraded until the link recovery is complete. These errors should be monitored closely as they typically affect multiple SAN devices.

I/O Traffic

I/O traffic results have different meanings for different operating systems. The Linux and UNIX operating systems bundle small block I/O into large 128 KB block requests, and performance at the upper end of the I/O block spectrum is an important concern. Microsoft Windows, on the other hand, defaults to a maximum I/O block of 64 KB and does not bundle small requests into larger ones.

Monitoring Fabric Switch Performance

Switch performance best practices should focus on the establishment of baselines. Use Aggregate Port and Port I/O metrics to establish typical IOPS rates and throughput rates as well as common error rates, average queue depths, and response times. Monitoring SAN switch and overall SAN performance primarily involves three metrics: IOPS (I/O operations per seconds), bandwidth, and latency.

Measuring IOPS and bandwidth can tell you how much work or activity is taking place in the SAN. Measuring latency tells you how effectively the SAN is doing its work, as well as whether the SAN is meeting its service objectives. By using switches and HBAs to view error rates, you can pinpoint the source of SAN performance problems. Error rates can include loss of signal or synchronization, re-transmissions, link failure, or invalid CRC.

Best Practices

Follow these best practices to optimize switch performance:

- Keep the highest performing directors at the core of the SAN.
- Connect storage devices and the highest performing applications to the core.
- Benchmark the performance on oversubscribed ports.
- Leave the Fibre Channel (FC) ports at auto-negotiate for host and storage connections.

Fibre Channel Port Types

Understanding FC port types can help you to identify ports along a storage path on which you are monitoring or collecting performance metrics. The following tables describe the different types of Fibre Channel ports:

Node Ports	Description
N_ port	Port on the node (such as, host or storage device) used with both FC-P2P or FC-SW topologies; also known as Node port.
NL_ port	Port on the node used with an FC-AL topology; also known as Node Loop port.
F_ port	Port on the switch that connects to a node point-to-point (for example, connects to an N_port); also known as Fabric port. An F_port is not loop capable
FL_ port	Port on the switch that connects to an FC-AL loop (such as, to NL_ports); also known as Fabric Loop port.
E_ port	Connection between two fibre channel switches. Also known as an Expansion port. When E_ports between two switches form a link, that link is referred to as an inter-switch link (ISL).
EX_ port	Connection between a fibre channel router and a fibre channel switch. On the side of the switch it looks like a normal E_port, but on the side of the router it is an EX_ port.
TE_ port	Cisco addition to Fibre Channel, now adopted as a standard. It is an extended ISL or EISL. The TE_port provides not only standard E_port functions but allows for routing of multiple VSANs (Virtual SANs). This is accomplished by modifying the standard Fibre Channel frame (vsan tagging) upon ingress/egress of the VSAN environment. The TE_port is also known as Trunking E_port.

General Ports	Description
Auto	Auto or auto-sensing port found in Cisco switches, can automatically become an E_, TE_, F_, or FL_port as needed.
Fx_port	Generic port that can become an F_port (when connected to a N_port) or a FL_ port (when connected to an NL_port). Found only on Cisco devices where over-subscription is a factor.
G_port	G_port or generic port on a switch that can operate as an E_port or F_port. The G_port is found on Brocade and McData switches.
L_port	Loose term used for any arbitrated loop port, NL_port or FL_port. L_port is also known as Loop port.
U_port	Loose term used for any arbitrated port. U_port is also known as Universal port and is found only on Brocade switches.

Switch Performance Metrics

Th following metrics are used to monitor switch performance.

Metric	Description	Units	Common Use
Aggregated Port Bytes Received	Sum of bytes received for all ports in a switch over an interval	MB/s	Measure inbound traffic for all ports on the switch.
Aggregated Port Bytes Transmitted	Sum of bytes transmitted for all ports in a switch over an interval	MB/s	Measure outbound traffic for all ports on the switch.
Bytes Received	Number of bytes received over a given interval	MB/s	Measure inbound traffic for specific ports on the switch.
Bytes Transmitted	Number of bytes transmitted over a given interval	MB/s	Measure outbound traffic for specific ports on the switch.
CRC Errors	Number of Cyclic Redundancy Check errors over a period of time	Errors	Isolate CRC errors on a specific initiator or between devices
Link Failures	Number of link Failures over a period of time	Failures	Isolate connection failures and the effect on performance

5 EMC Symmetrix Performance Metrics

This section describes the HP Storage Essentials performance metrics for the EMC Symmetrix devices running in your storage environment. It includes information about scheduling and collecting performance data (see Collecting EMC Symmetrix Performance Statistics below), and explains how performance metrics are derived (see About Counter Rollover on page 47).

For a list of performance management metrics, see EMC Symmetrix System Performance Metrics on page 51 and Transient Values for EMC Symmetrix Performance Metrics on page 62.

Collecting EMC Symmetrix Performance Statistics

If you purchased the HP Storage Essentials Performance Pack Enterprise for EMC Symmetrix license, HP Storage Essentials uses the internal Symmetrix SMI-S provider to request and receive Symmetrix performance data. This data is sent and received through remote communications to the **storsrvd** daemon running on the EMC Solutions Enabler server that HP Storage Essentials is currently using to communicate with the corresponding Symmetrix. HP Storage Essentials also requires the **storsrvd** daemon to be running for regular (non-performance) data collection.

All EMC Symmetrix statistical performance data collection, both scheduled and real time, is triggered using the HP Storage EssentialsPerformance Manager user interface. In the 9.4.0 version of HP Storage Essentials, the requests for Symmetrix performance data are serviced by the HP Storage Essentials internal Symmetrix SMI-S provider (not by the EMC Symmetrix SMI-S provider bundled with EMC Solutions Enabler).

Caution: It is important that the EMC Solutions Enabler **storstpd** daemon is NOT started or running while the HP Storage EssentialsPerformance Manager is using the same Solutions Enabler server to collect performance data. HP Storage Essentials will likely encounter problems collecting Symmetrix performance data if the **storstpd** daemon is running. See Scheduling EMC Symmetrix Performance Data Collection on next page.

Before You Collect Statistics

Check the HP Storage Essentials support matrix (available from the Documentation Center located in the top-level directory of the StorageEssentialsDVD) for the minimum EMC Solutions Enabler version requirements. Attempts to collect Symmetrix performance data using an unsupported Solutions Enabler server can result in errors.

Best Practices

Keep in mind the following recommended best practices when collecting performance data:

- Do not schedule collection using an unsupported version of the EMC Solutions Enabler server. This will result in errors.

- When possible, schedule collection frequencies between 15 and 30 minutes. Scheduled collection frequencies greater than 2 hours will not be serviced.
- HP does NOT recommend that you schedule a collection frequency of less than 5 minutes.
- Check the HP Storage Essentials cimom and appstorm logs when expected data is missing or errors are suspected.

Scheduling EMC Symmetrix Performance Data Collection

Due to limitations in the communications interface to the **storsrvd** daemon, only one request for data at a time can be made to a single Solutions Enabler server. For this reason, it is important that you schedule collection of Symmetrix performance data using the HP Storage Essentials Spread Start Time option (default) on the Data Collection - Edit Schedule page (**Configuration** > **Performance** > **Data Collection**).

This figure shows the Spread Start Time option on the Edit Schedules page on the Data Collection tab.

The amount of time to complete a single request can vary according to the number of volume statistics requested for a single Symmetrix and other factors. If the scheduled frequency (referred to as Interval) of collections (referred to as Runs) for Symmetrix Performance Data Collectors that are targeted at the same Solutions Enabler server is too short, collections could start backing up or erroring out.

Collection Frequency

The recommended frequency for Symmetrix data collection is between 15 and 30 minutes.

Data collection errors could begin surfacing for frequencies of less that 5 minutes, especially in larger environments. With the exception of the few statistics described as "point in time" in EMC Symmetrix System Performance Metrics on page 51, each EMC Symmetrix statistic value is averaged over the sample period between two successive collections. For example, if the scheduled collection frequency is 15 minutes, then a Read Data Rate value of 1,500 KB/s means that there was an average of 1,500 KB transferred per second between the time at which this statistic was collected and the previous collection ten minutes ago. Due to this averaging, scheduled frequencies of more than 1 hour tend to flatten the data in the line graphs, diminishing the granularity and value of the data represented by minimizing visual spikes and dips in the data rate between scheduled collections.

Although the scheduling interface allows input of frequencies for 2 hours or greater, the Performance Manager collection engine will fail to produce and store statistics for intervals greater than 2 hours, in which case, errors will be logged.

Real-time Data Collection

Real-time data collection in the Performance Manager is intended for short-term inspection of a few targeted statistics, and is not affected by, and is independent of, the previously mentioned schedules. Real-Time collectors are fixed at 20 second intervals. If expected Symmetrix performance data is missing or if errors are encountered while collecting real-time data, check both the cimom and the appstorm logs.

About Counter Rollover

When you calculate derived statistics, it is important to account for *counter rollover*. Counter rollover occurs where an accumulating performance counter in a computer system (for example, sequential_read_ios) reaches its maximum value and then restarts the count from zero.

A useful analogy is the odometer in your automobile. A counter rollover occurs when the odometer reaches its maximum value, rolls over to zero, and begins counting again. As a hypothetical example, if the maximum odometer count were 999,999 miles, when the car reaches 1,000,000 miles, the odometer would show 0 (zero) miles. If you calculated the miles traveled between mileage points, then you would calculate the delta between two readings taking into account the rollover. For example, if the maximum limit for the odometer is 999,999 miles, and the last reading was 900,000 and the current reading is 100,000, the change in the value since the last reading would be 200,000 miles.

In HP Storage Essentials, where possible, variables are derived by using the deltas or summations of the individual parts. This facilitates a more accurate calculation of the change in a counter value (for example, deltaReadIOs) in regards to handling a single rollover condition since the last time the counter value was read. You will find more information about this in the following topics, which describe how to account for counter rollover when calculating derived performance statistics:

- How Changes in Counter Values Are Calculated on next page
- How Rates Account for Single Rollover on page 49
- Duration Values Used to Calculate Rates on page 50
- Counters and KByte Conversion on page 50

How Changes in Counter Values Are Calculated

Typically, a change in value is calculated as `current - previous`; that is, the current value minus the previous value. In the case of a counter rollover, however, this formula produces an incorrect value. For example, if the current value is 100,000 and the previous value was 900,000, `current - previous` equals "– 800,000," which (due to counter rollover) is incorrect.

Therefore, when the current value is less than the previous value, an assumption is made that the counter has rolled over, and only rolled over one time. For this reason, when a rollover is detected, the change is calculated using the following formula:

```
delta = (1 + max_counter_value) - previous_value + current_value
```

The notation `(1 + max_counter_value)` can be expressed more simply as `bn`, where b is number base and n is the number of counter digits, referred to as num_counter_bits.

If we use the odometer example described in About Counter Rollover on previous page, the number base (given in decimal digits) is 10, and the number of digits is 6. Therefore, `b^n` is 10^6 and equals 1,000,000. This is the same value as `1 + max_counter_value`.

Here is how we derive the delta value:

```
b^n = 10^6 = 1,000,00

previous value = 900,000

current value = 100,000

1,000,000 - 900,000, + 100,000 = 200,000
```

This formula accounts for a single rollover condition and produces the correct delta value of 200,000 miles.

In the previous example, `b^n` is the same as `max_counter_value + 1`.

Calculating the Number Base

The number base for accumulating performance counters on computer systems is binary. As such, b=2 and num_counter_bits is usually 64 or 32.

In most cases, the `(1 + max_counter_value)` is calculated as follows:

```
b^n = 2^64 = 18,446,744,073,709,551,616

b^n = 2^32 = 4,294,967,296
```

For the Symmetrix performance metrics calculations, keep in mind the following.

The `num_counter_bits` is assumed to be 64 and, thus, `(1 + max_counter_value) = 2^64 = 18,446,744,073,709,551,616`.

Any other value for `num_counter_bits` is explicitly noted in this guide.

Any variable prefixed with "delta" implies the variable's value was set using the following logic:

```
if current_value < previous_value
```

```
delta = (1 + max_counter_value) - previous_value + current_value
else
delta = current - last
```

How Rates Account for Single Rollover

The following example uses the Total Read Rate metric to describe how rates account for single rollover.

The main variable used to calculate the Total Read Rate metric is deltaTotalReadIOs. When calculating the value for deltaTotalReadIOs, the deltas for both ReadIOsRandom and ReadIOsSequential are first calculated, where a single rollover in each can be accounted for. Next, the delta in TotalReadIOs is set equal to the sum of the individual parts. The calculations used in these steps are described in detail below.

Note: When calculating counter rollover, there is an assumption that a counter has not rolled over more than once.

Here are the steps for how a typical rate such as Total Read Rate is calculated:

1. Calculate delta and account for single rollover for random I/O (if applicable):

   ```
   deltaReadIOsRandom = current_ReadIOsRand - previous_ReadIOsRandom
   ```

2. Calculate delta and account for single rollover for sequential I/O (if applicable):

   ```
   deltaReadIOsSequential = current_ReadIOsSeq – previous_ReadIOsSeq
   ```

3. Calculate total read I/Os using previously derived delta values in steps 1 and 2:

   ```
   deltaTotalReadIOs = deltaReadIOsRandom + deltaReadIOsSequential
   ```

4. Calculate duration between collection points:

   ```
   duration_in_seconds = current_collection_time_counter - previous_
   collection_time_counter
   ```

5. Calculate total read rate using previously derived values in steps 3 and 4.

   ```
   TotalReadRate = deltaTotalReadIOs / duration_in_seconds
   ```

Example

The following example assumes that the Read IO counters have a maximum value of 999,999. When they reach the maximum value, the counter rolls over to zero (0). The values are returned as follows:

```
previous_collection_time_counter = 600,000
current_collection_time_counter = 600,100
previous_ReadIOsRandom = 200,000
current_ReadIOsRandom = 800,000
previous_ReadIOsSeq = 900,000
```

```
current_ReadIOsSeq = 200,000
```

Using the steps, this is how the Total Read Rate is calculated using the above values:

1. Calculate delta value for random read I/Os with no rollover:

   ```
   deltaReadIOsRandom: 800,000 - 200,000 = 600,000
   ```

2. Calculate the delta value for sequential read I/Os with rollover:

   ```
   deltaReadIOsSequential (rollover): (999,999 + 1) - 900,000 +
   200,000 = 300,000
   ```

3. Calculate the total read I/Os using the values derived in steps 1 and 2:

   ```
   deltaTotalReadIOs = 600,000 + 300,000 = 900,000
   ```

4. Calculate the value for duration:

   ```
   duration_in_seconds: 600,100 – 600,000 = 100
   ```

5. Calculate total read rate using values derived in steps 3 and 4:

   ```
   TotalReadRate = 900,000 IOs / 100 seconds = 900 ReadIOs / second
   ```

Duration Values Used to Calculate Rates

Duration is the elapsed time between the previous statistic collection and the current statistic collection. It is typically measured in seconds and used to calculate rates (such as ReadIOs per second).

The num_sym_timeslices counter is the Symmetrix time stamp in half-second units and represents the time at which the performance data statistics were collected and stored in the array. This counter provides for a more accurate calculation of duration than simply using the difference in system timestamps between the last collection and the current collection.

Where available, the EMC Symmetrix num_sym_timeslices counter is used to calculate the duration in seconds:

```
duration = deltaNumHalfSecTimeSlices / 2
```

For statistics that do not include the num_sym_timeslices counter, the system timestamp is used.

Counters and KByte Conversion

Most EMC Symmetrix counters that represent KBytes have been converted from "512 Byte" blocks to KBytes by dividing by 2 because there are two blocks per KByte.

Therefore, the max value, or point of rollover, for such counters in the database is half of the normal 64-bit max value used for the actual number of blocks returned to the counter by the Symmetrix array. This means that the num_counter_bits value is reduced by one.

To help you understand to which counters the above applies, the value of num_counter_bits is explicitly noted as either 63 or 31 in the metric descriptions in EMC Symmetrix System Performance Metrics on the facing page.

EMC Symmetrix System Performance Metrics

EMC Symmetrix system metrics are stored in the EMCSYMSTORAGESYSTEMSTATS and MVCS_EMCSYMSSSTATSVW database tables.

Metric data that is rolled up into hourly, daily, weekly, and monthly data points is stored in the corresponding EMCSYMSTORAGESYSTEMSTATS rollup tables and have a suffix of _H, _D, _ W, and _M respectively. For example, hourly collected data is stored in EMCSYMSTORAGESYSTEMSTATS_H and rolled up into daily data statistics stored in the EMCSYMSTORAGESYSTEMSTATS_D table.

The following table describes the performance metrics provided by HP Storage Essentials for EMC Symmetrix storage systems.

Metric	Description	Unit	Formula
Average Read Size	Average read size.	Bytes	AvgReadSize = (deltaKBytesRead x 1024) / deltaTotalReadIOsRandomAndSeq
Average Write Size	Average write size.	Bytes	AvgWriteSize = (deltaKBytesWritten x 1024) / deltaTotalWriteIOsRandomAndSeq
Deferred Write Rate	Rate of deferred write requesst. A deferred write is a write hit. A deferred write occurs when the I/O write operations are staged in cache and will be written to disk at a later time.	Req/s	DeferredWriteRate = deltaEMCDeferredWriteIOs / duration
Delayed DFW Rate	Delayed DFW request rate. A delayed deferred fast write (DFW) is a write-miss. A delayed DFW occurs when the I/O write operations are delayed because the system or device write-pending limit was reached and the cache had to de-stage slots to the disks before the writes could be written to cache.	Req/s	DelayedDfwRate = deltaEMCDelayedDFWIOs / duration

Metric	Description	Unit	Formula
Max Pending Flush Limit	Maximum number of write-pending slots for the entire Symmetrix. System write-pending limit is equal to 80% of the available cache slots. Symmetrix write-pending limit is not simply a sum of all Symmetrix device write-pending slots. It depends on other factors such as cache size and the Symmetrix configuration.System property. This is a point-in-time value captured at the time the statistics are taken.	Bytes	MaxPendingFlushLimit = EMCMaxKBPendingFlush x 1024
Pending Flush	Number of tracks in cache that are waiting to be de-staged to disk and cannot be overwritten. This is a point-in-time value captured at the time the statistics are taken.	Bytes	PendingFlush = EMCKBPendingFlush x 1024
Pending Format	Number of format pending tracks. This count can be less than the last-taken statistic; it is a point-in-time value captured at the time the statistics are taken.	Bytes	PendingFormat = EMCKBPendingFormat x 1024
Percent Hits	Ratio of total hits (random and sequential) to total I/Os (random and sequential).	%	PctHitIOs = 100 x (deltaTotalHitIOsRandomAndSeq / deltaTotalIOsRandomAndSeq)
Percent Read Hits	Read cache hit ratio (percentage of read hits).[2]	%	PctReadHitIOs = 100 x (deltaReadHitIOsTotal / deltaTotalReadIOsRandomAndSeq)
Percent Reads	Ratio of read I/Os to total I/Os.	%	PctReadIOs = 100 x (deltaTotalReadIOsRandomAndSeq / deltaTotalIOsRandomAndSeq
Percent Reads Seq[1]	Sequential read rate. (percentage of sequential reads to TotalIOs including SequentialReads).[2]	%	PctSeqReadIOs = 100 x (deltaReadIOsSeq / deltaTotalReadIOsRandomAndSeq)
Percent Write Hit I/Os	Write cache hit ratio (percentage of write hits).[2]	%	PctWriteHitIOs = 100 x (deltaWriteHitIOsTotal / deltaTotalWriteIOsRandomAndSeq)

Metric	Description	Unit	Formula
Percent Writes	Ratio of write I/Os to total I/Os.	%	PctWriteIOs = 100 x (deltaTotalWriteIOsRandomAndSeq / deltaTotalIOsRandomAndSeq)
Prefetch Data Rate	Rate of prefetched Bytes per second.	Bytes/s	PrefetchRate = (deltaEMCKBPrefetched * 1024) / duration
Read Data Rate	Read throughput rate (Bytes per second).	Bytes/s	ReadDataRate = (deltaKBytesRead x 1024) / duration
Read Hits	Read cache hit rate.	Req/s	ReadHitRate = deltaReadHitIOsTotal / duration
Read Hits Seq	Rate of read cache hits per second (sequential hits only).	Req/s	SeqReadHitRate = deltaReadHitIOsSeq / duration
Read Rate Random	Random read cache request rate (requests per second).	Req/s	ReadRate = deltaReadIOs / duration
Read Rate Seq	Sequential read rate.[2]	Req/s	SeqReadRate = deltaReadIOsSeq / duration
Read Rate Total	Read request rate that includes both random and sequential reads.	Req/s	ReadRateTotal = deltaTotalReadIOsRandomAndSeq / duration
Total Data Rate	Total bytes read and written per second.	Bytes/s	TotalDataRate = (deltaKBytesTransferred x 1024) / duration
Total I/O Rate	I/O rate which includes random and sequential reads and writes.	Req/s	TotalIORate = deltaTotalIOsRandomAndSeq / duration
Write Data Rate	Write throughput rate (Bytes per second).	Bytes/s	WriteDataRate = (deltaKBytesWritten x 1024) / duration
Write Flush Data Rate	Number of tracks (expressed in Bytes) written per sec from cache to disks.	Bytes/s	WriteFlushRate = (deltaEMCWriteKBytesFlushed x 1024) / duration
Write Hits	Write cache hit rate.	Req/s	WriteHitRate = deltaWriteHitIOsTotal / duration
Write Hits Seq	Rate of write cache hits per second (sequential hits only).	Req/s	SeqWriteHitRate = deltaWriteHitIOsSeq / duration
Write Rate Total	Write cache request rate (requests per second) that includes both random and sequential wites.	Req/s	WriteRate = deltaTotalWriteIOsRandomAndSeq / duration

Metric	Description	Unit	Formula
Write Rate Seq	Write cache request rate (requests per second) and includes only sequential writes.	Req/s	SeqWriteRate = deltaWriteIOsSeq / duration

[1] The SYM CLI is calculating setPctSeqReadIOs as ratio of ReadSeq to Total Reads. The EMC Perf Metrics glossary defines this as ReadsSeq to TotalIOs.

[2] This metric does NOT apply to Symm3 data.

EMC Symmetrix Volume Performance Metrics

EMC Symmetrix volume metrics are stored in the EMCSYMVOLUMESTATS and MVCS_ EMCSYMVOLUMESTATSVW database tables.

Metric data that is rolled up into hourly, daily, weekly, and monthly data points is stored in the corresponding EMCSYMVOLUMESTATS rollup tables having a suffix of _H, _D, _W, and _M respectively. For example, the EMCSYMVOLUMESTATS_W table contains the rolled-up weekly performance data statistics for EMC Symmetrix volumes.

The following table describes the HP Storage Essentials performance metrics for volumes connected to EMC Symmetrix storage systems.

Metric	Description	Unit	Formula
Average I/O Size	Average size of an I/O operation performed by the Symmetrix device.	Bytes	AvgIOSize = (deltaKBytesTransferred x 1024) / deltaTotalIOsRandomAndSeq
Average Read Size	Average size of a read I/O operation performed by the Symmetrix device.	Bytes	AvgReadSize = (deltaKBytesRead x 1024) / deltaTotalReadIOsRandomAndSeq
Average Write Size	Average size of a write I/O operation performed by the Symmetrix device.	Bytes	AvgWriteSize = (deltaKBytesWritten x 1024) / deltaTotalWriteIOsRandomAndSeq
Max Write Pending Threshold	Maximum number of write-pending slots available (expressed in Bytes) for the Symmetrix device.[1]	Bytes	MaxWritePendingThreshold = current_ EMCMaxKBPendingFlush x 1024

Metric	Description	Unit	Formula
Pending Flush[2]	Number of cache slots (expressed in Bytes) that were write pending for the logical volume at a point in time. This number changes according to the cache de-stage activity rate and the number of writes. A write is pending when it has been written to cache but has not yet been written to the disk.	Bytes	PendingFlush = current_ EMCKBPendingFlush x 1024
Percent Hits	Percentage of I/O cache hit operations performed by the Symmetrix device that were immediately satisfied by cache.	%	PctHitIOs = 100 x (deltaTotalHitIOsRandomAndSeq / deltaTotalIOsRandomAndSeq)
Percent Misses	Percentage of read and write operations performed by the Symmetrix device that were misses.	%	PctMissIOs = 100 - PctHitIOs
Percent Read Hits Random	Ratio of read hit I/Os to Total I/Os.	%	PctReadHitIOsRandom = 100 x (deltaReadHitIOs / deltaTotalIOsRandomAndSeq)
Percent Read Hits Total	Percentage of read cache hit I/Os (including both random and sequential) operations performed by the Symmetrix device.	%	PctReadHitIOsTotal = 100 x (deltaReadHitIOsTotalRandomAndSeq / deltaTotalReadIOsRandomAndSeq)
Percent Reads	Percentage of read I/O operations performed by the Symmetrix device.	%	PctReadIOs= 100 x (deltaTotalReadIOsRandomAndSeq / deltaTotalIOsRandomAndSeq)
Percent Read Misses Random	Ratio of read cache miss I/Os to Total I/Os.	%	PctReadMissIOsRandom = 100 x (deltaReadMissIOsRandom / deltaTotalIOsRandomAndSeq)
Percent Read Misses Total	Percentage of read cache miss I/O operations performed by the Symmetrix device.	%	PctReadMissIOsTotal = 100 x (deltaReadMissIOsTotalRandomAndSeq / deltaTotalReadIOsRandomAndSeq)

Metric	Description	Unit	Formula
Percent Write Hits	Percentage of cache write hit I/O operations performed by the Symmetrix device.	%	PctWriteHitIOs = 100 x (deltaWriteHitIOsTotalRandomAndSeq / deltaTotalWriteIOsRandomAndSeq)
Percent Writes	Percentage of total write I/O operations performed by the Symmetrix device.	%	PctWriteIOs = 100 x (deltaTotalWriteIOsRandomAndSeq / deltaTotalIOsRandomAndSeq)
Percent Write Misses	Percentage of write I/O operations performed by the Symmetrix device that were write misses.	%	PctWriteMissIOs = 100 x (deltaWriteMissIOsTotalRandomAndSeq / deltaTotalWriteIOsRandomAndSeq)
Read Data Rate	Number of Bytes read by the Symmetrix device each second.	Bytes	ReadDataRate = (deltaKBytesRead x 1024) / duration
Read Hit Rate Random	Number of random read hit I/O operations performed each second by the Symmetrix device. The read hits per sec metric for the Symmetrix device statistic does not include sequential read hits. In contrast, the Read Hit Rate Total metric includes random and sequential read hits per second.	Req\s	ReadHitRateRandom = deltaReadHitIOs / duration
Read Hit Rate Seq	Number of sequential read hit requests performed (per second) by the Symmetrix device.	Req/s	SeqReadHitRate = deltaReadHitIOsSeq / duration
Read Hit Rate Total	Total number of read hit operations (random and sequential) performed each second by the Symmetrix device.	Req/s	ReadHitRateTotal = deltaReadHitIOsTotalRandomAndSeq / duration

Metric	Description	Unit	Formula
Read Rate Random	Number of I/O operations performed each second by the Symmetrix device that were random reads. This Random Reads per sec metric for the Symmetrix device statistic does not include sequential reads. In contrast, the Read Rate Total metric includes random and sequential read hits per second.	Req/s	ReadRateRandom = deltaReadIOs / duration
Read Rate Seq	Number of sequential read I/O operations performed each second by the Symmetrix device.	Req/s	SeqReadRate = deltaReadIOsSeq / duration
Read Rate Total	Read request rate including both random and sequential read operations performed each second by the Symmetrix device.	Req/s	ReadRateTotal = deltaTotalReadIOsRandomAndSeq / duration
Sampled Average Read Time	Completion time of a read as measured by the host director. Measurements are taken for a sample set of approximately 30% of the I/Os.	MS	SampledAvgReadTimeMs = curr.getEMCSampledReadsTime(), curr.getEMCSampledReads(), null
Sampled Average Write Time	Completion time of a write as measured by the host director. Measurements are taken for a sample set of approximately 30% of the I/Os.	MS	SampledAvgWriteTimeMs = current_ EMCSampledWritesTime / current_ EMCSampledWrites
Total Data Rate	Total Bytes read and written per second.	Bytes/s	TotalDataRate = deltaKBytesTransferred x 1024) / duration

Metric	Description	Unit	Formula
Total Hit Rate	Total number of I/O operations (random and sequential) performed each second by the Symmetrix device that were immediately satisfied by cache.	Req/s	TotalHitRate = readHitRateTotalRandomAndSeq + writeHitRateTotalRandomAndSeq
Total I/O Rate	Total number of read I/O and write I/O operations (random and sequential) performed each second by the Symmetrix device.	Req/s	TotalIORate = readRateTotalRandomAndSeq + writeRateTotalRandomAndSeq
Total I/O Rate Random	Number of I/O operations performed each second by the Symmetrix device, including writes and random reads. In contrast, the Total IO Rate metric includes writes, random reads, and sequential reads.	Req/s	TotalIORateRandom = deltaTotalIOsRandom / duration
Total Miss Rate	Total number of I/O operations (random and sequential) performed each second by the Symmetrix device that were NOT immediately satisfied by cache	Req/s	TotalMissRate = TotalIORate - TotalHitRate
Write Data Rate	Number of Bytes written by the Symmetrix device each second.	Bytes/s	WriteDataRate = (deltaKBytesWritten * 1024) / duration
Write Hit Rate Total	Number of write hit operations (random and sequential) performed each second by the Symmetrix device.	Req/s	WriteHitRate = deltaWriteHitIOsTotalRandomAndSeq / duration
Write Hit Rate Seq	Number of sequential write hits that occurred each second for the Symmetrix device.	Req/s	SeqWriteHitRate = deltaWriteHitIOsSeq / duration

Metric	Description	Unit	Formula
Write Miss Rate	Number of write misses that occurred for the Symmetrix device each second.	Req/s	WriteMissRate = deltaWriteMissIOsTotalRandomAndSeq / duration
Write Rate Random	Write cache request rate (requests per second) that includes only random writes by the Symmetrix device.	Req/s	WriteRate = deltaWriteIOs / duration
Write Rate Seq	Number of sequential write I/O operations performed each second by the Symmetrix device.	Req/s	SeqWriteRate = deltaWriteIOsSeq / duration
Write Rate Total	Write cache request rate (requests per second) including both random and sequential I/Os performed for the Symmetrix device.	Req/s	WriteRateTotal = deltaTotalWriteIOsRandomAndSeq / duration

[1] Max Write Pending Threshold is not a static number. It depends on Symmetrix activity. Each Symmetrix device is assigned a limit of write-pending slots that can dynamically change between a base value and a value three times the base (the maximum value). When the Max Write Pending Threshold reaches three times the base value, writes to the device are delayed so that the cache can destage, which frees the cache slots. As cache slots are freed, the writes resume. While the write-pending limit is reached, disk directors operate in a priority destage write mode. This gives write data higher priority than normal. During the delay, writes to this Symmetrix device are counted as write misses.

[2] Certain HP Storage Essentials metrics, such as Pending Flush, are shown in bytes, whereas their corresponding EMC metric are shown in tracks. The corresponding EMC metric for Pending Flush is EMC Num WP Tracks. The Tracks-to-KB conversion factor is based on the Enginuity version of the EMC Symmetrix array as follows: for Enginuity < 5771: Tracks-to-KB = 32; for Enginuity >= 5771: Tracks-to-KB = 64 Num Tracks = (number of Bytes / 1024) / Tracks-to-KB.

EMC Symmetrix Front-end Controller Performance Metrics

EMC Symmetrix front-end controller metrics are stored in the EMCSYMFECONTROLLERSTATS and MVCS_EMCSYMFECONTSTATSVW database tables.

Metric data that is rolled up into hourly, daily, weekly, and monthly data points is stored in the corresponding EMCSYMFECONTROLLERSTATS roll-up tables having a suffix of _H, _D, _W, and _M respectively. For example, monthly rolled up data is stored in the EMCSYMFECONTROLLERSTATS_M database table.

The following table describes the performance metrics provided by HP Storage Essentials for EMC Symmetirx front-end controllers.

Metric	Description	Unit	Formula
Device Write Pending Event Rate	Number of times each second that the write-pending limit for a specific Symmetrix device was reached. When the limit is reached, additional write I/O operations are deferred while waiting for data in cache to be destaged to the disk.	Events/s	DeviceWritePendingEventRate = deltaEMCDeviceFlushPendingEvents / duration
Percent Hits	Percentage of requests performed by the host director and immediately satisfied by cache.	%	PctHitIOs = 100 x (deltaEmcTotalHitIOs / deltaTotalIOs)
Percent Reads	Percentage of read requests performed by the host director.	%	PctReadIOs = 100 x (deltaReadIOs / deltaTotalIOs)
Percent Writes	Percentage of write requests performed by the host director over the sample interval.	%	PctWriteIOs = 100 x (deltaWriteIOs / deltaTotalIOs)
Read Rate	Number of random read requests performed each second by Symmetrix host director.	Req/s	ReadRate = deltaReadIOs / duration
Slot Collision Rate	Number of slot collisions each second. A slot collision occurs when two or more directors try to access the same cache slot and the slot happens to be locked for an update operation by one of the directors.	Collisions/s	SlotCollisionRate = deltaEMCSlotCollisions / duration
System Write Pending Event Rate	Number of times each second that write activity was heavy enough to use up the system limit set for write tracks occupying cache. When the limit is reached, writes are deferred until data in cache is written to disk.	Events/s	SystemWritePendingEventRate = deltaEMCSystemFlushPendingEvents / duration

Metric	Description	Unit	Formula
Total Data Rate	Number of Bytes transferred through the Symmetrix Director each second.	Bytes/s	TotalDataRate = (deltaKBytesTransferred x 1024) / duration
Total Hit Rate	Number of read and write requests performed each second by the host director that was immediately satisfied by cache.	Req/s	TotalHitRate = deltaEMCTotalHitIOs / duration
Total I/O Rate	Number of I/O operations performed each second by the Symmetrix host director. This metric represents activity between the Symmetrix device and the host or SAN device.	Req/s	TotalIORate = deltaTotalIOs / duration
Write Rate	Number of write requests performed each second by the host directors.	Req/s	WriteRate = deltaWriteIOs / duration

EMC Symmetrix Host FC Port Performance Metrics

EMC Symmetrix Front-End Controller metrics are stored in the EMCSYMHOSTFCPORTSTATS and MVCS_EMCSYMHOSTFCPTSTATSVW database tables.

Metric data that is rolled up into hourly, daily, weekly, and monthly data points is stored in the corresponding EMCSYMHOSTFCPORTSTATS roll-up tables having a suffix of _H, _D, _W, and _M respectively. For example the daily roll-up performance statistics are stored in the EMCSYMHOSTFCPORTSTATS_D database table.

The following table describes the performance metrics provided by HP Storage Essentials for EMC Symmetrix host Fibre Channel (FC) ports.

Metric	Description	Unit	Formula
Average I/O Size	Average number of Bytes transferred through the Symmetrix host port per I/O operation.	Bytes/s	AvgIOSize = (deltaKBytesTransferred x 1024) / deltaTotalIOs
Total Data Rate[1]	Number of Bytes transferred through the Symmetrix host port each second.	Bytes/s	TotalDataRate = (deltaKBytesTransferred x 1024) / duration

Metric	Description	Unit	Formula
Total I/O Rate	Number of I/O operations performed each second by the Symmetrix host port. This metric represents activity between the Symmetrix device and the host or SAN device.	Req/s	TotalIORate = deltaTotalIOs / duration

[1] For this metric the **`num_counter_bits`** value equals 31.

Transient Values for EMC Symmetrix Performance Metrics

The following transient variables are used to calculate rates for the EMC Symmetrix performance metrics. These variables are pre-fixed with "delta" and use the current–previous logic described in this section.

Logic for Delta Variables

Any variable prefixed with “delta” that is calculated as `current-previous` implies that the variable value was set using the following logic:

```
if current_value < previous_value

delta = (1 + max_counter_value) - previous_value + current_value

else

delta = current - last
```

In this instance, (1 + max_counter_value) is equal to 2^num_counter_bits. The num_counter_bits value is 64 unless stated otherwise.

In all formulas, the term "duration" is defined as follows:

```
duration = deltaNumHalfSecTimeSlices / 2
```

Transient Values for EMC Symmetrix Systems

The following delta variables are used for EMC Symmetrix system performance rates. The Formula column shows the equation used to derive the value of the variable on the left.

Variable Name	Formula
deltaEMCDelayedDFWIOs	currentEMCDelayedDFWIOs - previousEMCDelayedDFWIOs
deltaEMCDeferredWriteIOs	currentEMCDeferredWriteIOs – previousEMCDeferredWriteIOs

Variable Name	Formula
deltaEMCKBPrefetched	current_EMCKBPrefetched - previous_ EMCKBPrefetched
deltaEMCWriteKBytesFlushed	currentEMCWriteKBytesFlushed - previousEMCWriteKBytesFlushed
deltaKBytesRead*	current_KBytesRead - previous_KBytesRead
deltaKBytesTransferred	deltaKBytesRead + deltaKBytesWritten
deltaKBytesWritten*	current_KBytesWritten - previous_KBytesWritten
deltaReadHitIOs	current_ReadHitIOs - previous_ReadHitIOs
deltaReadHitIOsSeq	current_EMCSequentialReadHitIOs - previous_ EMCSequentialReadHitIOs
deltaReadHitIOsTotal	deltaReadHitIOs + deltaReadHitIOsSeq
deltaReadIOs	current_ReadIOs - previous_ReadIOs
deltaReadIOsSeq	current_EMCSequentialReadIOs - previous_ EMCSequentialReadIOs
deltaTotalHitIOsRandomAndSeq	deltaReadHitIOsTotal + deltaWriteHitIOsTotal
deltaTotalIOsRandom**	current_TotalIOs - previous_TotalIOs
deltaTotalIOsRandomAndSeq	deltaTotalReadIOsRandomAndSeq + deltaTotalWriteIOsRandomAndSeq
deltaTotalReadIOsRandomAndSeq	deltaReadIOs + deltaReadIOsSeq
deltaTotalWriteIOsRandomAndSeq	deltaWriteIOs + deltaWriteIOsSeq
deltaWriteHitIOs	current_WriteHitIOs - previous_WriteHitIOs
deltaWriteHitIOsSeq	current_EMCSequentialWriteHitIOs - previous_ EMCSequentialWriteHitIOs
deltaWriteHitIOsTotal	deltaWriteHitIOs + deltaWriteHitIOsSeq
deltaWriteIOs	current_WriteIOs - previous_WriteIOs
deltaWriteIOsSeq	current_EMCSequentialWriteIOs - previous_ EMCSequentialWriteIOs

*The value of num_counter_bits is 63 for calculating this delta value. Blocks are converted to KB by dividing by 2, therefore use 63 counter bits for value when roll-over occurs. See Logic for Delta Variables above.

**The total_io counter holds the sum of random reads plus random writes.

Transient Values for EMC Symmetrix Volumes

The following table lists the delta variables used for EMC Symmetrix volume performance rates. The Formula column shows the equation used to derive the value of the variable on the left.

Variable Name	Formula
deltaKBytesRead*	current_KBytesRead - previous_KBytesRead
deltaKBytesTransferred	deltaKBytesRead + deltaKBytesWritten
deltaKBytesWritten*	current_KBytesWritten - previous_ KBytesWritten
deltaReadHitIOs	current_ReadHitIOs - previous_ReadHitIOs
deltaReadHitIOsSeq	current_EMCSequentialReadHitIOs - previous_ EMCSequentialReadHitIOs
deltaReadHitIOsTotalRandomAndSeq	deltaReadHitIOs + deltaReadHitIOsSeq
deltaReadIOs	current_ReadIOs - previous_ReadIOs
deltaReadIOsSeq	current_EMCSequentialReadIOs - previous_ EMCSequentialReadIOs
deltaReadMissIOsRandom	deltaReadIOs - deltaReadHitIOs
deltaReadMissIOsSeq	deltaReadIOsSeq - deltaReadHitIOsSeq
deltaReadMissIOsTotalRandomAndSeq	deltaReadMissIOsRandom + deltaReadMissIOsSeq
deltaTotalHitIOsRandomAndSeq	deltaReadHitIOsTotalRandomAndSeq + deltaWriteHitIOsTotalRandomAndSeq
deltaTotalIOsRandom**	current_TotalIOs - previous_TotalIOs
deltaTotalIOsRandomAndSeq	deltaTotalReadIOsRandomAndSeq + deltaTotalWriteIOsRandomAndSeq
deltaTotalReadIOsRandomAndSeq	deltaReadIOs + deltaReadIOsSeq
deltaTotalWriteIOsRandomAndSeq	deltaWriteIOs + deltaWriteIOsSeq
deltaWriteHitIOs	current_WriteHitIOs - previous_WriteHitIOs
deltaWriteHitIOsSeq	current_EMCSequentialWriteHitIOs - previous_ EMCSequentialWriteHitIOs
deltaWriteHitIOsTotalRandomAndSeq	deltaWriteHitIOs + deltaWriteHitIOsSeq

Variable Name	Formula
deltaWriteIOs	current_WriteIOs - previous_WriteIOs
deltaWriteIOsSeq	current_EMCSequentialWriteIOs - previous_EMCSequentialWriteIOs
deltaWriteMissIOsRandom	deltaWriteIOs - deltaWriteHitIOs
deltaWriteMissIOsSeq	deltaWriteIOsSeq - deltaWriteHitIOsSeq
deltaWriteMissIOsTotalRandomAndSeq	deltaWriteMissIOsRandom - deltaWriteMissIOsSeq

*The value of num_counter_bits is 63 for calculating this delta value. Blocks are converted to KB by dividing by 2; therefore, use 63 counter bits for value when roll-over occurs. See Logic for Delta Variables on page 62.

**The total_io counter holds the sum of random reads plus random writes.

Transient Values for EMC Symmetrix Front-end Controllers

The delta variables are used for EMC Symmetrix front-end controller performance rates. The Formula column shows the equation used to derive the value of the variable on the left.

Variable Name	Formula
deltaKBytesTransferred	current_KBytesTransferred – previous_KBytesTransferred
deltaReadIOs	current_ReadIOs - previous_ReadIOs
deltaWriteIOs	current_WriteIOs - previous_WriteIOs
deltaTotalIOs	current_TotalIOs – previous_TotalIOs
deltaEMCTotalHitIOs	current_EMCTotalHitIOs - previous_EMCTotalHitIOs
deltaEMCSlotCollisions	current_EMCSlotCollisions - previous_EMCSlotCollisions
deltaEMCSystemFlushPendingEvents	current_EMCSystemFlushPendingEvents - previous_EMCSystemFlushPendingEvents
deltaEMCDeviceFlushPendingEvents	current_EMCDeviceFlushPendingEvents - previous_EMCDeviceFlushPendingEvents

Transient Values for EMC Symmetrix Host FC Ports

The following delta variables are used for EMC Symmetrix host Fibre Channel (FC) port performance rates. The Formula column shows the equation used to derive the value of the variable on the left. The num_counter_bits for calculating the KBytesTransferred for host FC ports is 31.

Variable Name	Formula
deltaKBytesTransferred	current_KBytesTransferred – previous_KBytesTransferred
deltaTotalIOs	current_TotalIOs – previous_TotalIOs

6 HP StorageWorks EVA Performance Metrics

HP Storage Essentials enables deep visibility into the performance levels of HP StorageWorks Enterprise Virtual Arrays (EVA) storage arrays by providing metrics for the following EVA storage infrastructure components:

- EVA Storage System Metrics on page 70
- EVA Controller Metrics on page 72
- EVA Physical Disk Metrics on page 74
- EVA Host FC Port Metrics on page 76
- EVA Volume Metrics on page 79

Click the metric set above that interests you to view a table that describes the statistics collected for that array component.

Metric Table Conventions

You will find the following conventions used in EVA performance statistics tables listed above:

- To simplify formulas, the value Time represents the difference in seconds between the most recent two StatisticTime values returned from the SMI-S provider.
- Items marked with an asterisk (*) indicate they are duplicates of SMI-S statistics.
- All statistic names used in shown formulas are listed in a separate "raw statistics" table for each array component. See references provided in the introduction to each table.

Collecting EVA Performance Metrics

This section describes recommendations and best practices for collecting performance metrics for HP StorageWorks Enterprise Virtual Array (EVA) systems in your environment.

As a first step, read the "Managing Performance Data Collection" and "Viewing Performance Data" in the *HP Storage Essentials Storage Performance Management Guide* to understand data collection for HP EVA and which features are available for your system depending on installed products and licensing.

No performance metrics are gathered by HP Storage Essentials during Get Details (GAED). Running the Get Details process takes time, and therefore should not be run when you need to monitor array performance or collect performance statistics.

EVAPerf must be running before you can obtain performance statistics for EVA arrays.

Finding EVA Bottlenecks Using Performance Metrics

HP Storage Essentials metrics for the HP EVA array system can help you find bottlenecks by enabling you to see the performance throughput (I/O) rates and response times (latency). Bottlenecks are characterized by unexpected latencies, which may be due to heavy data workloads and resource contention. Identifying bottlenecks early can be an ongoing challenge for storage administrators. The following sections describe a few of the indicators and the HP Storage Essentials metrics that provide the information you need to manage the performance of your array.

I/O Per Second

To determine transmission loads (for managing load balancing and increased bandwidth needs), determine the total I/O per seconds (IOPS) over a time interval. An IOPS is 1000 ms/CommandTime + SeekTime + SettledownTime +DataTransferTime.

Read Latency

Chart the total array MB/s, total array requests, and FP port latencies as a baseline for normal performance of your EVA array. When looking for bottlenecks, check the average time it takes to complete a read request (from initiation to information receipt) from the physical disks (for all volumes in the EVA array) by looking at the Total Volume Avg Read Miss Latency (Sec). If the latency rate increases (rule of thumb: 30 ms or more), and there is no correlated increase in IOPS or MB/s, you should investigate further. Although it is acceptable for certain environments to run at >30 ms, you should base what you consider to be an acceptable latency rate on your specific environment design and workload profile.

Write Latency

Check the average time it takes to complete a write request (from initiation to receipt of write completion) for all volumes by selecting the Total Volume Avg Write Latency (sec). If the latency is 5 ms or more, this indicates a potential bottleneck in the system and you should investigate further.

Write latency may also increase due to increased write activity to the controller. As the total write activity to the controller increases, flush rates will become higher and steadier. Check your flush rates using the Total Volume Flush Data Rate (Bytes/Sec) metric. 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.

Remote data replication using HP Continuous Access EVA might also be the cause of write latencies, especially if Sync replication is enabled. Hosts which access their LUNs through the replication ports can also be negatively affected.

Queue Depth

Disk queue depth can indicate whether the workload asked of the physical disk is greater than the intended design. In such cases, you might consider adding more physical disks.

FP queue depths can be caused by array latency, as well as not balancing hosts for fair access. For example, a host taht has a queue depth of 128 can starve a host that has a queue depth of 8.

Monitoring EVA Performance

To optimize performance of your HP Enterprise Virtual Array (EVA), follow these best practices:

- Establish a performance baseline. Collect metrics that represent normal performance conditions for your system. You can then use this baseline to compare future metrics and analyze issues.
- Track IOPS rates and throughput.
- Track error rates, average queue depth and response times.
- To maximize single array performance: Fill the EVA with as many disk drives as possible.
- To maximize application performance, you should perform a performance analysis before adding any significant application load to an existing EVA.
- Configure as few Disk Groups as possible.
- When creating Disk Groups, as a best practice use equal capacity disks.
- Where different physical disk sizes are used in a Disk Group, set up workloads so that the larger physical disks house a proportionately higher amount of the data and that workloads do not exceed design capacities.
- To provide the highest performance, use 15K-RPM disks. This is a guideline for performance management and not a remedy for availability.
- For high I/O write workloads, to reduce the utilization of the mirror port, migrate the high I/O intensive Vdisk from VRAID 5 to VRAID 1.
- The only implication of having different physical disk sizes in a disk group is that the larger physical disks will house a proportionally higher amount of the data. If physical disks are not asked to do more than they are designed to do, then no harm is done.
- Always leave read caching enabled.
- Attempt to balance the workload demand evenly across both controllers on an EVA.
- Although both controllers in an EVA can access all physical disks, a Vdisk is assigned to be managed by only one controller at a time. Because the Vdisks can be accessed by either controller, all FP ports can be used. ALUA aware drivers will use the owning controller's FP ports for the read data and all available ports for the write access.
- Enable a common subset of metrics for standard collection. These might include CPU utilization (%), total I/O rate (MB/s), total requests per second, and FP read and write latencies. If you create a baseline that includes these metrics, you can then quickly determine 1) if the latency increases and 2) if there is an increased workload that is responsible for it. Consider placing these metrics into a basic EVA template.

- Enabling all collectors, or more than are necessary, can result in decreased Command View EVA responsiveness and intermittent performance data collection failure.

EVA Storage System Metrics

The metrics track performance for the HP StorageWorks Enterprise Virtual Array (EVA). Storage system metrics are stored in the HPEVASTORAGESYSTEMSTATS database table. For descriptions of statistics in formulas, see EVA Storage System Raw Statistics on page 72 and the EVA Volume Raw Statistics on page 81.

Metric	Description	Units	Formula
Total Data Rate*	Rate data can be transmitted between devices for the entire storage system.	Bytes/s	(Δ KBytesTransferred x 1024) / Δ Time
Total I/O Rate*	Average number of I/O operations in requests per second for both sequential and non-sequential reads and writes for the entire storage system	Req/s	Δ TotalIOs / Δ Time
Total Volume Avg Read Hit Latency	Average time to complete a read request (from initiation to information receipt) from the array cache memory for all volumes in the array.	Sec	(Δ ReadHitLatency / 1000) / Δ ReadHitIOs
Total Volume Avg Read Miss Latency	Average time to complete a read request (from initiation to information receipt) from the physical disks for all volumes	Sec	(Δ ReadMissLatency / 1000) / Δ ReadMissIOs
Total Volume Avg Read Size	Average data read size for all volumes	Bytes	(Δ KBytesRead x 1024) / Δ ReadIOs
Total Volume Avg Write Latency	Average time to complete a write request (from initiation to receipt of write completion) for all volumes	Sec	(Δ KBytesTransferred x 1024) / Δ Time
Total Volume Avg Write Size	Average write size for all volumes	Bytes	(Δ KBytesWritten x 1024) / Δ WriteIOs

Metric	Description	Units	Formula
Total Volume Data Rate	Rate data can be transmitted between devices for all volumes	Bytes/s	(Δ KBytesTransferred x 1024) / Δ Time
Total Volume Flush Data Rate	Rate at which data is written to physical disks in array	Bytes/s	(Δ FlushKBytes x 1024) / Δ Time)
Total Volume Flush Rate	Aggregate of all flush counters: mirror flush, cache flush, host writes to snapshots and snapclones	Bytes/s	Δ FlushRequests / Δ Time
Total Volume I/O Rate	Average number of I/O operations per second for both sequential and non-sequential read and write operations for all volumes	Req/s	Δ TotalIOs / Δ Time
Total Volume Mirror Data Rate	Rate at which data travels across the mirror port to complete read and write requests to all virtual disks	Bytes/s	(Δ MirrorKBytes x 1024) / Δ Time
Total Volume Pct Read IOs	Percentage (%) of read I/O operations per second for both sequential and non-sequential reads for all volumes	%	100 x (Δ ReadIOs / Δ TotalIOs)
Total Volume Pct Write IOs	Percentage (%) of write I/O operations per second for both sequential and non-sequential writes for all volumes	%	100 x (Δ WriteIOs / Δ TotalIOs)
Total Volume Prefetch Data Rate	Rate data is read from the physical disk to cache in anticipation of subsequent reads when a sequential stream is detected	Bytes/s	(Δ PrefetchKBytes x 1024) / Δ Time)
Total Volume Read Data Rate	Rate data is read from the virtual disk by all hosts and includes transfers from the source array to the destination array	Bytes/s	(Δ KBytesRead x 1024) / Δ Time
Total Volume Read Hit Data Rate	Rate at which data is read from the array cache memory because of read hit requests	Bytes/s	(Δ ReadHitKBytes x 1024) / Δ Time)

Metric	Description	Units	Formula
Total Volume Read Hit Rate	Number of read requests per second completed from the array cache memory	Req/s	Δ ReadHitIOs / Δ Time
Total Volume Read Miss Data Rate	Rate at which data is read from physical disks because the data was not present in the array cache memory	Bytes/s	(Δ ReadMissKBytes x 1024) / Δ Time
Total Volume Read Miss Rate	Number of read requests (per second) that were not available from cache memory and therefore were completed from the physical disks instead	Req/s	Δ ReadMissRequests / Δ Time
Total Volume Read Rate	Number of read requests per second completed from a virtual disk that were sent to all hosts	Req/s	Δ ReadIOs / Δ StatisticTime
Total Volume Write Data Rate	Rate at which data is written to the virtual disk by all hosts, including transfers from the source array to the destination array	Bytes/s	Δ KBytesWritten x 1024) / Δ Time
Total Volume Write	Number of write requests per second completed to a virtual disk that were received from all hosts	Req/s	Δ WriteIOs / Δ Time

*This metric is a duplicate of an SMI-S statistic.

EVA Storage System Raw Statistics

HP Storage Essentials collects the following storage system array statistics and stores them in the HPEVASTORAGESYSTEMSTATS database table.

Raw Statistic	Type	Units
Total I/Os*	uint64	Count
Kbytes Transferred*	uint64	KB

*This metric is a duplicate of an SMI-S statistic.

EVA Controller Metrics

For EVA storage systems in which HP Storage Essentials supports performance data collection, the following performance metrics are monitored for EVA controllers. See the EVA Controller Raw Statistics on page 74 for descriptions of the statistics shown in the formulas.

Metric	Description	Units	Formula
Average Read Sizea*	Amount of data read (per second) from physical disk	Bytes	(Δ KBytesRead x 1024) / Δ ReadIOs
Average Write Size*	Amount of data written (per second) to physical disks	Bytes	(Δ KBytesWritten x 1024) / Δ WriteIOs
CPU Percent**	Percentage (%) of time that the central processing unit on the controller is active. A completely idle controller shows 0%. A controller saturated with activity shows 100%.	%	100 x (Δ CpuBusyCounter / Δ StatisticsTime)
Data Transfer Percent**	Similar to % Processor Time except that it does not include time for internal processes not related to host-initiated data transfers	%	100 x (Δ DataTxCounter / Δ StatisticsTime)
Percent Reads*	Percentage (%) of CPU time dedicated to reads	%	100 x (Δ ReadIOs / Δ TotalIOs)
Percent Writes*	Percentage (%) of CPU time dedicated to writes	%	100 x (Δ WriteIOs / Δ TotalIOs)
Read Data Rate*	Rate at which data is read from the controller by all disks	Bytes/s	(Δ KBytesRead x 1024) / Δ Time
Read Latency	Average time it takes to complete a read request (from initiation to receipt of write completion) through the controller	Sec	(Δ ReadLatency / 1000) / Δ ReadIOs
Reads*	Rate at which data is read from each host port	Req/s	Δ ReadIOs / Δ Time
Total Data Rate*	Rate at which data can be transmitted between devices for the controller	Bytes/s	(Δ KBytesTransferred x 1024) / Δ Time
Total I/O Rate*	Average number of I/O operations as requests per second for both sequential and non-sequential reads and writes for the controller	Req/s	Δ TotalIOs / Δ Time
Write Data Rate*	Rate at which data is written to the virtual disk by all hosts and includes transfers from the source array to the destination array	Bytes/s	(Δ KBytesWritten x 1024) / Δ Time
Write Latency	Average time it takes to complete a write request (from initiation to receipt of write completion)	Sec	(Δ WriteLatency / 1000) / Δ WriteIOs

Metric	Description	Units	Formula
Writes*	Number of write requests per second completed to a virtual disk that were received from all hosts	Req/s	Δ WriteIOs / Δ Time

*This metric is a duplicate of an SMI-S statistic.

**This metric is collected as both an HP Storage Essentials and an EVAPerf performance statistic.

EVA Controller Raw Statistics

The following values are collected by HP Storage Essentials and, except for the CpuCounter and DataTxCounter, are aggregated by the provider from the underlying port metrics. All values are stored in the HPEVACONTROLLERSTATS database table.

Raw Statistic	Type	Units
CpuCounter	uint64	Ms
DataTxCounter	uint64	Ms
Total I/Os*	uint64	Count
Kbytes Transferred*	uint64	KB
Read I/Os*	uint64	Count
KBytesRead*	uint64	KB
Write I/Os*	uint64	Count
KBytesWritten*	uint64	KB
Read Latency	uint32	ms
Write Latency	uint32	ms

*This metric is a duplicate of an SMI-S statistic.

EVA Physical Disk Metrics

This following metrics track performance statistics for EVA physical disks. These values are stored in the HPEVADISKSTATS database table. For descriptions of the statistics shown in formulas, see EVA Physical Disk Raw Statistics on page 76 .

Metric	Description	Units	Formula
Average Read Size*	Amount of data read from physical disk	Bytes	(Δ KBytesRead x1024) / Δ ReadIOs

Metric	Description	Units	Formula
Average Write Size*	Amount of data written to physical disk	Bytes	(Δ KBytesWritten x 1024) / Δ WriteIOs
Drive Latency metric**	Average time to complete read/write requests from the physical disk drive	Sec	(Δ DriveLatency / 1000) / Δ TotalIOs
Percent Reads*	Percentage (%) of CPU time dedicated to reads	%	100 x (Δ ReadIOs / Δ TotalIOs)
Percent Writes*	Percentage (%) of CPU time dedicated to writes	%	100 x (Δ WriteIOs / Δ TotalIOs)
Queue**	Average number of outstanding requests against the physical disk	Req	Δ DriveQueueDepth / Δ Statistic Time
Read Data Rate***	Rate at which data is read from the virtual disk by all hosts, including transfers from the source array to the destination array	Bytes/s	(Δ KBytesRead x 1024) / Δ Time
Read Latency**	Average time to complete a read request (from initiation to information receipt) from the array volume	Sec	(Δ ReadLatency / 1000) / Δ ReadIOs
Reads***	Rate at which data is read from each host port	Req/s	Δ ReadIOs / Δ Time
Total Data Rate*	Rate at which data can be transmitted between devices for the host port	Bytes/s	(Δ KBytesTransferred x 1024) / Δ Time
Total I/O Rate*	Average number of I/O operations (requests per second) for both sequential and non-sequential reads and writes for the host port	Req/s	Δ TotalIOs / Δ Time
Write Data Rate***	Rate at which data is written to the virtual disk by all hosts, including transfers from the source array to the destination array	Bytes/s	(Δ KBytesWritten x 1024) / Δ Time
Write Latency**	Average time to complete a write request (from initiation to receipt of write completion)	Sec	(Δ WriteLatency / 1000) / Δ WriteIOs
Writes***	Number of write requests per second completed to a virtual disk that were received from all hosts	Req/s	(Δ KBytesRead x 1024) / Δ Time

*This metric is a duplicate of an SMI-S statistic.

**This metric is collected as both an HP Storage Essentials and an EVAPerf performance statistic.

***This metric is collected by both HP Storage Essentials and EVAPerf and is a duplicate of an SMI-S statistic.

EVA Physical Disk Raw Statistics

The following values are collected by HP Storage Essentials and stored in the HPEVADISKSTATS database table.

Raw Statistic	Type	Units
TotalIOs*	uint64	Count
KBytesTransferred*	uint64	KB
ReadIOs*	uint64	Count
KBytesRead*	uint64	KB
WriteIOs*	uint64	count
KBytesWritten*	uint64	KB
DriveQueueDepth	uint64	Count
DriveLatency	uint64	ms
ReadLatency	uint64	ms
WriteLatency	uint64	ms

*This metric is a duplicate of an SMI-S statistic.

EVA Host FC Port Metrics

For EVA storage systems in which HP Storage Essentials supports performance data collection, the following items are monitored for EVA Host FC Ports. See EVA Host FC Port Raw Statistics on page 78 for descriptions of the statistics shown in the formulas.

Metric	Description	Units	Formula
Average Read Size	Amount of data read (per second) from physical disks	Bytes	(Δ KBytesRead x1024) / Δ ReadIOs
Average Write Size	Amount of data written (per second) to physical disks	Bytes	(Δ KBytesWritten x 1024) / Δ WriteIOs

Metric	Description	Units	Formula
Bad CRC	Number of bad CRC errors. Indicates that the Cyclic Redundancy Check (CRC) which compares a data stream against a stored checksum, has found the data stream changed and therefore no longer reliable. Use to help the transmitter detect errors in the frame that are caused by bad writes, bad media, damaged links/hardware, excessive link errors, and transfer rates.	Count	–
Bad Receive Characters	Number of bad receive characters in the bit stream. Use to determine the number of bad frames associated with the Bad CRC metric above.	Count	–
Discard Frames	Number of frames discarded due to Bad CRCs. Frames are the basic unit of communication between two N_ports, and are composed of a starting delimiter, header, payload, CRC, and end delimiter.	Count	–
Link Fail	Number of link failures. Use to find issues with the fiber optic cable or transceiver or the SAN infrastructure.	Count	–
Loss of Signal	Number of times the receiver reports loss of signal. Indicator that fiber optic signal no longer exists. Use to assist in troubleshooting signal loss.	Count	–
Loss of Sync	Number of times the receiver logic reports loss of sync has timed-out. Use to determine the number of times an intermittent loss of synchronization in communication signals was received by an enclosure connected to a Fibre Channel (FC) loop.	count	–
Percent Reads	Percentage (%) of CPU time dedicated to reads.	%	100 x (Δ ReadIOs / Δ TotalIOs)
Percent Writes	Percentage (%) of CPU time dedicated to writes	%	100 x (Δ WriteIOs / Δ TotalIOs)
Protocol Error	Number of errors in the protocol between the channel and the control unit. Use to differentiate between protocol errors and link errors.	count	–

Metric	Description	Units	Formula
Queue	Average number of outstanding host requests against all virtual disks accessed through this host port	Count	Δ QDepth / Δ Time
Read Data Rate	Rate at which data is read from the controller by all disks	Bytes/s	(Δ KBytesRead x 1024) / Δ Time
Read Latency**	Average time to complete a read request (from initiation to receipt of write completion) through the controller	Sec	(Δ ReadLatency / 1000) / Δ ReadIOs
Reads**	Rate at which data is read from each host port	Req/s	Δ ReadIOs / Δ Time
Receive Abnormal End of Frame	Number of times a bad frame was detected during data transmission	Count	–
Total Data Rate*	Rate in which data can be transmitted between devices for the host port	Bytes/s	(Δ KBytesTransferred x 1024) / Δ Time
Total I/O Rate*	Average number of I/O operations as requests per second for both sequential and non-sequential reads and writes for the host port	Req/s	Δ TotalIOs / Δ Time
Write Data Rate**	Rate at which data is written to the virtual disk by all hosts and includes transfers from the source array to the destination array	Bytes/s	(Δ KBytesWritten x 1024) / Δ Time
Write Latency**	Average time to complete a write request (from initiation to receipt of write completion)	Sec	(Δ WriteLatency / 1000) / Δ WriteIOs
Writes**	Number of write requests per second completed to a virtual disk that were received from all hosts	Req/s	Δ WriteIOs / Δ Time

*This metric is a duplicate of an SMI-S statistic.

**This metric is collected as both an HP Storage Essentials and EVAPerf performance statistic.

EVA Host FC Port Raw Statistics

The following values are collected by HP Storage Essentials and stored in the HPEVAHOSTFCPORTSTATS database table.

Raw Statistic	Type	Units
TotalIOs*	uint64	Count

Raw Statistic	Type	Units
KbytesTransferred*	uint64	KB
ReadIOs	uint64	count
KBytesRead	uint64	KB
ReadLatency	uint32	ms
WriteIOs	uint64	Count
KBytesWritten	uint64	KB
WriteLatency	uint32	ms
FlushRequests	uint64	Count
QDepth	uint64	Count

*This metric is a duplicate of an SMI-S statistic.

EVA Volume Metrics

The following metrics track performance for HP StorageWorks Enterprise Virtual Array (EVA) volumes. See EVA Volume Raw Statistics on page 81 for descriptions of the statistics shown in the formulas.

Metric	Description	Units	Formula
Average Read Size*	Amount of data read (per second) from physical disks	Bytes	(Δ KBytesRead x 1024) / Δ ReadIOs
Average Write Size*	Amount of data written (per second) to physical disks	Bytes	(Δ KBytesWritten x 1024) / Δ WriteIOs
Flush Data Rate	Rate at which data is written to a physical disk for the associated virtual disk	Bytes/s	(Δ FlushKBytes x 1024) / Δ Time)
Flushes	Number of cache flush requests per second	Req/s	Δ FlushRequests / Δ Time
Mirror Data Rate	Rate at which data travels across the mirror port to complete read and write requests for the associated virtual disk	Bytes/s	(Δ MirrorKBytes x 1024) / Δ Time
Percent Reads*	Percentage (%) of CPU time dedicated to reads	%	100 x (Δ ReadIOs / Δ TotalIOs)

Metric	Description	Units	Formula
Percent Writes*	Percentage (%) of CPU time dedicated to writes	%	100 x (Δ WriteIOs / Δ TotalIOs)
Prefetch Data Rate	Rate at which data is read from the physical disk to cache in anticipation of subsequent reads when a sequential stream is detected	Bytes/s	(Δ PrefetchKBytes x 1024) / Δ Time)
Read Data Rate*	Rate at which data is read from the virtual disk by all hosts, including transfers from the source array to the destination array	Bytes/s	(Δ KBytesRead x 1024) / Δ Time
Read Hit Data Rate**	Rate at which data is read from the array cache memory because of read hit requests	Bytes/s	(Δ ReadHitKBytes x 1024) / Δ Time)
Read Hit Latency**	Average time to complete a read request (from initiation to information receipt) from the array volume	Sec	(Δ ReadHitLatency / 1000) / Δ ReadHitIOs
Read Hits***	Number of read requests (per second) completed from the array cache memory	Req/s	Δ ReadHitIOs / Δ Time
Read Miss Data Rate**	Rate at which data is read from physical disks because the data was not present in the array cache memory	Bytes/s	(Δ ReadMissKBytes x 1024) / Δ Time
Read Miss Latency**	Average time it takes to complete a read request (from initiation to information receipt) from the physical disks for all volumes	Sec	(Δ ReadMissLatency / 1000) / Δ ReadMissIOs
Read Misses**	Number of read requests (per second) that failed to complete from the array cache memory and were completed from physical disks instead	Req/s	Δ ReadMissRequests / Δ Time
Reads*	Rate at which data is read from each host host port	Req/s	Δ ReadIOs / Δ Time
Total Data Rate*	Rate at which data can be transmitted between devices for the host port	Bytes/s	(Δ KBytesTransferred x 1024) / Δ Time
Total I/O Rate***	Average number of I/O operations in requests per second for both sequential and non-sequential reads and writes for the hostport	Req /s	Δ TotalIOs / Δ Time

Metric	Description	Units	Formula
Write Data Rate***	Rate at which data is written to the virtual disk by all hosts and includes transfers from the source array to the destination array	Bytes/s	(Δ KBytesWritten x 1024) / Δ Time
Write Latency**	Average time to complete a write request (from initiation to receipt of write completion)	Sec	(Δ WriteLatency / 1000) / Δ WriteIOs
Writes***	Number of write requests per second completed to a virtual disk that were received from all hosts	Req /s	Δ WriteIOs / Δ Time

*This metric is a duplicate of an SMI-S statistic.

**This metric is collected as both an HP Storage Essentials and an EVAPerf performance statistic.

***This metric is a duplicate of an SMI-S statistic and is collected as both an HP Storage Essentials and an EVAPerf performance statistic.

EVA Volume Raw Statistics

The following values are collected by HP Storage Essentials and stored in the HPEVAVOLUMESTATS database table.

Raw Statistic	Type	Units
Total I/Os*	uint64	Count
Kbytes Transferred*	uint64	KB
Read I/Os*	uint64	Count
Read Hit I/Os*	uint64	Count
KBytesRead*	uint64	KB
Write I/Os*	uint64	Count
KBytesWritten*	uint64	KB
Read Hit KBytes	uint32	KB
Read Hit Latency	uint32	Ms
Read Miss Requests	uint64	Count
Read Miss Kbytes	uint64	KB
Read Miss Latency	uint32	ms
Write Latency	uint32	ms

Raw Statistic	Type	Units
Flush Requests	uint64	Count
Flush Kbytes	uint64	KB
Mirror KBytes	uint64	KB
Prefetch KBytes	uint64	KB

*This metric is a duplicate of an SMI-S statistic.

7 HP StorageWorks XP and HDS Array Performance Metrics

Performance Manager enables you to drill down deep into an XP or HDS array and view all component performance metrics. These include cache and shared memory statistics, CHA processor and port statistics for front-end controllers, and DKA processor statistics for back-end controllers, volume statistics for storage pools, and RAID group statistics for array groups.

HP Storage Essentials provides performance metrics for the following storage infrastructure components for HP XP and HDS arrays. Click a metric set that interests you in the following list to see the statistics collected for that component.

- XP and HDS Array Storage System Metrics on page 89
- XP and HDS Array Group Metrics on page 90
- XP and HDS Array Volume Metrics on page 92
- XP and HDS Front-end Controller CLPR Metrics on page 95
- XP and HDS Front-end Controller (CHA) Metrics on page 96
- XP and HDS MPB Controller Metrics on page 97

Back-end controller DKA metrics are provided for HP XP arrays only. See XP Back-end Controller (DKA) Metric on page 96.

XP Array Performance Tools

HP Storage Essentials is the only performance tool for XP arrays that displays a full SAN topology, showing you all devices along the connection path from application to disk volume. Using the Performance Manager you can generate performance charts for each component in the path for which data has been collected and generate reports which show performance statistics.

To fully use Performance Manager capabilities, use the tool in collaboration with other HP Storage Essentials components. Use the Discovery features of HP Storage Essentials to discover each element in your SAN environment. You can then display these discovered elements in a SAN topology path that clearly shows the connections from hosts to arrays. Select the performance metrics you want to monitor using Performance Manager and when you are ready, generate reports for hosts, switches, and arrays.using the Report Optimizer.

Depending on your IT practices, you could also be using one of the following HP StorageWorks XP performance tools:

- **XP Command View AE** enables you to configure common tasks for hosts, switches, and arrays;manages multiple XP disk arrays; uses the Remote Web Console to manage one XP disk array at a time.

- **XP Performance Advisor** enables you to monitor hosts, applications running on the hosts, and switches and arrays connected to the hosts; focuses on performance management of XP disk arrays; provides reporting, performance estimates, and a history database.

You will find certain instructions for the Remote Web Console feature of XP Command View in this guide. For the most part, instructions are specific to HP Storage EssentialsPerformance Manager.

About Setting Up XP and HDS Array Metric Collection

HP XP and Hitachi Data Systems (HDS) performance data collection works by collecting data in-band from a command device. You need to set up at least one host in the SAN with a command device from the array, with a CIM Extension installed. The CIM Extension automatically detects the command device. After running Discovery Get All Element Details, view the array's properties, then click **Edit Proxy Host** to select the host you want used to collect performance data.

Metrics for Hitachi Data Systems (HDS) arrays are only available when the Command Device is running on a Microsoft Windows host.

License Setup for Performance Pack

The Performance Pack Enterprise plug-in is required for performance collection of HP XP arrays with HP Storage Essentials. You can view XP array data in Performance Manager only if you are appropriately licensed.

The Performance Pack Enterprise license provides the ability to collect and report additional performance data for specified XP arrays. The number of required licenses depends upon the number of XP arrays you want to include for additional collection and reporting.

You must complete a Get Details for XP arrays before you import the license for the Performance Pack. After you import the license, you can start the data collectors from the Performance Data Collection page (**Configuration** > **Performance** > **Data Collection**). Although XP and HDS arrays are displayed after you run Discovery, you must run a Get Details for the collectors to run properly.

For more information about licensing XP and HDS arrays, see the *HP Storage Essentials Storage Performance Management Guide*.

Collecting XP and HDS Array Performance Metrics

Before you begin to collect performance metrics, be sure to read the "Viewing Performance Data" in the *HP Storage Essentials Storage Performance Management Guide*. It describes the Performance Manager and the data collection methods for HP XP and HDS arrays available for your system.

Performance metrics for XP and HDS arrays include global resources such as cache and shared memory, front-end controllers, XP back-end controllers, ports, storage pools and their constituent volumes, Thin Provisioning/Snapshot Pools and V-VOLs, and the array groups.

The data collection process might take a considerable amount of time to complete and display in the Performance Manager. In certain cases, this might be as long as a couple of hours. Performance Manager requires two points to plot the first data point, and this affects the collection time interval along with the size of your storage array, the collector setup, and other related factors.

The minimum collection interval that can be set for the XP and HDS performance data collectors is one minute. However, collecting at frequencies less than five minutes, increases the amount of data in the database. HP recommends that you use an interval between 15 minutes and one hour. Optionally, set the collection interval lower while analyzing a problem, then restore the collector interval to the default. The interval for real time collection is always 20 seconds.

Non-performance data can be collected from the array using either the CV SMI-S provider or the built-in XP provider. However, you are required to use the built-in provider to collect back-end controller statistics. You will only see front-end controllers in the Performance Manager display when you connect to the array in Step 1 Discovery using CV SMI-S provider. Be sure to use the built-in provider in Step 1 Discovery to collect Back-end Controller statistics. See the *HP Storage Essentials User Guide* for instructions about how to do this.

When collecting performance statistics for XP logical array groups, be aware that a parity group is divided into several logical RAID groups. Thus, for example, parity group X-X can contain logical array groups X-X-1, X-X-2, and so forth. The collecting of performance statistics for logical array groups is only done at the parity group level. Because the Performance Manager for XP array statistics does not show the parity group, all the statistics are gathered at the first logical RAID group only (in our example, X-X-1), and no statistics are gathered at X-X-2 and the other subsequent logical RAID groups.

Finding XP and HDS Array Bottlenecks Using Performance Metrics

The following sections describe considerations when examining performance metrics for HP XP and HDS arrays.

When determining the cause of a performance bottleneck, it is important to observe how an array utilizes its memory and cache. Selecting an XP or HDS array element allows you to view shared memory and cache usage statistics for the array. The cache statistics are itemized per Cache Logical Partition (CLPR).

To troubleshoot and isolate factors that could affect your application performance:

1. Navigate to the host running your application and display performance metrics by clicking it. You can click on either the application name in the path tree or application icon in the topology map.
2. Check the host processor utilization and physical/virtual memory consumption. If your processor is running near 100% and virtual memory consumption is greater than physical memory, the problem is most likely with the host not having enough resources to process the I/O for the application.
3. Select the application path where the application is running, and expand its members.

4. Check the LDEV read/write response time. The LDEV may have an extremely high response time (above 20ms) or zero response. When the response time is higher than expected, there may be a HP RESTRICTED resource contention problem within the array. In the case of zero response, there could be a link failure.
5. Chart the HBA, switches, and array port Bytes Transmitted/Received metrics on the same graph to verify that traffic is passing through to the array without problems. If you see large differences between the HBA, switch and array port, the problem could be a bad SFP or fiber link.
6. Chart the array port and LDEV Total Data Rate. The array port should have at least the same data rate as the LDEV. If the array port data rate is lower than LDEV that could mean you have too much write pending in cache.

Best Practices

As I/O is running, view the Cache Writes Pending for the CLPR (to which your host/application LUN(s) belong) in real time. To do this, select Real Time from the Period drop down. Write Pending indicates the amount of I/O waiting to be de-staged to disk. As Cache Writes Pending approaches 70 percent, typically the array begins to hold back I/O, which in turn affects your host response time. Some of the causes for excessive write pending could be too much I/O from other devices in the CLPR, replication processing, and external storage (unless you have cache write-through mode enabled).

Monitoring XP Baseline Performance

The following are recommendations for monitoring your XP baseline performance monitoring. These are useful as guides for general thresholds in HP Storage EssentialsPolicy Manager as well.

See the *HP Storage Essentials Storage Performance Management Guide* for more information about setting performance policies and alerts.

Front-End Port Usage (CPU Load)

HP recommends an average load under 50 percent. At 50 percent or more, the I/O response time gets doubled. Set the Policy Manager to send Info-Msg at 60% and Alert at 70%.

Back-End ACP Pair Usage (CPU Load)

Recommendations for back-end ACP pair usage are the same as front-end port usage. Set Policy Manager to send Info-Msg at 60% and Alert at 70%. This parameter is not available in the HP Storage Essentials XP Perf Policy. Also for 99 percent of XP customers, back-end CPU is not a performance issue.

Cache Utilization

HP recommends a threshold below 70 percent. If this value is greater than 70 percent, you should check the cache characteristics of your XP. Set Policy Manager to send Alert if greater than 70 percent.

Write Pending Rate

The write pending rate is very important. The value of the Percent Write Pending Data (%) is normally stable around 30%. If the percentage reaches 50 percent, you want to be alerted immediately. Set your Policy Manager to sent Alerts if it is >40% to >50%.

Write Response Time

The write response time indicated by the Write Response Time (Sec) for XP array volumes should be below 10ms (also in sync-CA environment). The baseline for this parameter is important. Set your Policy to send Alert if the response time is greater than 10 milliseconds (ms).

Array Group Utilization

HP recommends that you maintain an average load of under 50 percent. Set Policy Manager to send Info-Msg at >60% and Alert at >70%.

LDEV Utilization

The probability that one logical device (LDEV) has more than 70 percent of the load of an array group is very low. If this situation is true, you need to analyze manually. The overall significance of this metric value is low, as it is more important to look into the overall array group utilization.

Viewing External Storage and Continuous Access Link Performance

Currently, you must use the Remote Web Console (RWC) to determine which ports are used for Continuous Access and/or External Storage connectivity. Plot the ports you suspect may have a failure, and the switches used to interconnect the arrays. The port charts will show you how well the port is being utilized, and the switch chart will show any link failures between the arrays.

To diagnose your links:

1. Navigate to front-end controllers and chart the IOPS from initiator or target Port. If you want to see performance from all arrays simultaneously you will need to launch another browser session for each array.
2. Chart the back-end processor (XP DKA) metrics for the external arrays.
3. Chart cache write pending for all arrays. Your XP or HDS disk array should have cache passthrough enabled for external storage; therefore, you will see minimum impact on the CLPR metrics assigned to the external LUNs.
4. Monitor the switches to determine the link failures. You will need to launch another browser session to view switches and arrays simultaneously.
5. Assuming there are no link failures, determine which external array connections are excessively utilized by checking the cache writes pending versus link IOPS. If cache writes

pending on the external array is low and link IOPS are high, you could add another external link path to improve performance.

Optimizing Your XP and HDS Array Performance

RAID group utilization will let you know what kind of I/O is affecting performance. Some disk types and RAID levels work faster with Sequential versus Random I/O, and/or Reads versus. Writes. The storage admin should optimize the array to satisfy the customer's Service Level Agreement (SLA) while minimizing the cost of the array. The user will also need to know the origin of the I/O (that is, the volumes in use), the data path, and array resource usage.

To optimize performance on your HP XP array:

1. Chart RAID group utilization metrics and check to see what kind of I/O is most prevalent.
2. Determine which volumes are connected to the RAID group and chart their I/O type and response times.
3. Verify that the response times meet your Service Level Agreement.
4. Determine which volumes are candidates for optimization based on their I/O type and response times.
5. Check the ports to which the volumes are mapped and graph their throughput.
6. Verify that the port is not a bottleneck based on its maximum rated throughput versus its actual throughput.
7. Determine the host to which the volumes are mapped, and make sure its resources are not excessively utilized.
8. Determine which applications are using the volumes in the RAID group.
9. Locate new volumes to use as candidates for migration.
10. Check the candidate's current RAID Group utilization to verify that resources are available.

Viewing Pool and Group Performance

THP/Snapshot/Continuous Access Journal Pool Performance

XP Pools are composed of LDEVs from RAID Groups. To fully analyze your Pool, you must determine the RAID Groups to which the LDEVs belong. Best practice for creating a Pool is to dedicate the entire RAID Group to that Pool, which makes the performance more predicable, because no external source of IO and disk utilization will affect the Pool.

To diagnose your Pool:

1. Select Storage Pools and right click on the storage pool in question, then choose **Show Properties**. Select the pool volumes to determine the RAID Groups they belong to.
2. Chart the V-VOL performance from the storage pool.

3. Navigate to Array Groups and find the RAID Groups to which the Pool LDEVs belong. Chart the RAID Group utilization.
4. If your RAID Group utilization is high, focus your search down to I/O type. Chart Percent Reads/Writes for the RAID Group(s) and V-VOL(s) connected to the Pool.
5. Look for patterns in I/O type compared to V-VOL usage to determine which V-VOLs are most responsible for RAID Group utilization.

XP and HDS Array Storage System Metrics

Selecting an XP or HDS array element in Performance Manager allows you to view shared memory and cache usage statistics. The cache statistics are itemized per Cache Logical Partition (CLPR). It is important for the user to observe how the array is using its memory and cache in order to help determine the root cause of a performance bottleneck.

De-staging activities in a data storage system are controlled by providing a write pending list of elements, where each element is defined to store information related to a cache memory data element for which a "write to storage" is pending, and maintaining the write pending list so that de-staging of a data element can be based on the maturity of the pending write. Use the Write Pending Data (MBtyes)metric in the following table to monitor this information.

Array Storage System Metrics[1]

The following metrics measure the performance of HP XP or HDS arrays, including the P9500/XP storage arrays. See footnotes for exceptions.

Metric	Description	Units
CHA Cache Memory Busy Rate*	Rate at which the CHA cache is busy de-staging data to the DKA. Available for HP XP arrays only.	%
CHA Shared Memory Busy Rate*	Rate at which the CHA shared memory is busy	%
Cache Usage	Cache utilization in megabytes	MB
DKA Cache Memory Busy Rate*	Rate at which the DKA cache is busy de-staging data to the disk. Available for HP XP arrays only.	%
DKA Shared Memory Busy Rate*	Rate at which the DKA shared memory is busy. Available for HP XP arrays only.	%
Percent Cache Usage**	Cache utilization percent	%

Metric	Description	Units
Percent Sidefile Usage**	Percent utilization of the sidefile. A sidefile is an internal buffer that saves a copy of the data to be transmitted to a remote XP array. Use to track continuous access (CA) sidefile cache utilization and the potential impact of DR activities. Formula: SidefileUsage/CacheSize	%
Percent Write Pending Data**S	Percentage of pending writes based on the percentage of cache being used to buffer writes on the selected controller. Use to determine if a CLPR is needed or if attention needs to be directed toward journal parity groups. Formula: WritePendingData/CacheSize	%
Read Hits**	Read I/O requests per second satisfied from cache	Req/s
Sidefile Usage**	Sidefile cache utilization in megabytes	MB
Write Pending Data**	Indicator of pending writes based on cache in megabytes used to buffer writes on the selected controller.	MB

[1]Performance metrics in this table are not available for XP P9500 storage arrays discovered using Command View Advanced Edition (CVAE). For more information, see the Licensing chapter in the *Installation Guide*.

*Metric not available for XP and HDS RAID700 arrays. XP and HDS RAID700 arrays use MPB controllers.

**The default Cache Logical Partition (CLPR) used for obtaining performance information is CLPR 0. The default MP for the XP controller is MP 0.

XP and HDS Array Group Metrics

The Array Groups section lists all of the RAID Groups in the array. Select an array group to view its performance (for example, Utilization). You can also distinguish the I/O types and determine how the RAID group is being used to determine whether data should be migrated to another group.

Best Practices

Simultaneously graph the percentage of reads and writes, random and sequential I/O, average read and write size, and total utilization. By comparing the percentage of reads and writes, I/O types, and utilization, you can see how the RAID group is being used. If there is excessive random access, it could affect your sequential performance. You might, therefore, move your LUNs used for random I/O to another RAID group; for example, by using Auto LUN or Business Copy.

Array Group Metrics[1]

The following performance metrics are available for HP XP and HDS array groups and includes the P9500/XP storage arrays (see footnote for exceptions). The size of I/Os being issued affects disk latency; that is, large I/O sizes typically result in slightly higher latency.

Metric	Description	Units
Average Read Size	Average number of reads in bytes to the array group.	Bytes
Average Write Size	Average number of writes in bytes to the array group	Bytes
Percent Read Hits	Percentage (%) of cache reads for the array group	%
Percent Read Hits Random	Percentage (%) of random read I/O's that were de-staged from cache	%
Percent Read Hits Seq	Percentage (%) of sequential read I/O's that were de-staged from cache	%
Percent Reads	Percentage (%) of reads from cache	%
Percent Writes	Percentage (%) of writes from cache	%
Read Data Rate	Rate data is read from the array group by all hosts and includes transfers from the source array to the destination array	Bytes/s
Read Data Rate Random	Rate at which random data is read from the array group by all hosts	Bytes/s
Read Data Rate Seq	Rate at which sequential data is read from the array group by all hosts and includes transfers from the source array to the destination array	Bytes/s

Metric	Description	Units
Read Hits	Cache read hits in requests per second	Req/s
Read Response Time	Time required to complete a read I/O in seconds	Sec
Reads	Number of read I/O's in requests per second	Req/s
Total Data Rate	Rate at which data can be transmitted between devices for the selected array group	Req/s
Total I/O Rate	Total number of read or write operations taking place per second for the selected array group	Req/s
Utilization	Percentage of time disks in the array group are busy	%
Write Data Rate	Rate at which data is written to the array group by all hosts and includes transfers from the source array to the destination array	Bytes/s
Write Data Rate Random	Rate at which random data is written to the array group by all hosts	Bytes/s
Write Data Rate Seq	Rate at which sequential data is written to the array group by all hosts and includes transfers from the source array to the destination array	Bytes/s
Write Response Time	Time required to complete a write I/O in seconds	Sec
Writes	Number of write I/O's in requests per second	Req/s

[1]Performance metrics in this table are not available for XP P9500 storage arrays discovered using Command View Advanced Edition (CVAE). For more information, see the Licensing chapter in the *Installation Guide*.

XP and HDS Array Volume Metrics

Volume performance is monitored by selecting the storage pool containing the volume. The pool shows all volumes (LDEVs, LUSEs, and V-VOLs) associated with it. Select individual volumes to view their performance metrics. When you display volume metrics, you can view its response time to the host and data rate.

Best Practices

When diagnosing a performance bottleneck, first graph the application's volume response time and data rates. If response time is high, and data rate is low, then there might be a problem with any of the stages in between (switch, CHA, DKA, and so forth). If response time is low/normal and data rate is high, storage is not the source of your application's performance problem. If the application is using more than one LUN, you can graph metrics that have the same units on the same graph, but not in real time.

Array Volume Metrics[1]

The following metrics are available for HP XP and HDS array volumes, including the P9500/XP storage array volumes (see footnotes for exceptions). To simplify formulas, the value Time represents the difference in seconds between the most recent two StatisticTime values returned from the provider.

Metric	Description	Units	Formulas
Average Read Size	Average number of reads in bytes to the array group	Bytes	(Δ RandomReadData + Δ SequentialReadData) / (Δ RandomReadIOs + Δ SequentialReadIOs)
Average Write Size	Average number of writes in bytes to the array group	Bytes	(Δ RandomWriteData + Δ SequentialWriteData) / (Δ RandomWriteIOs + Δ SequentialWriteIOs)
Percent Read Hits	Percentage (%) of cache reads for the array group	%	100 x (Δ RandomReadHitIOs + Δ SequentialReadHitIOs) / (Δ RandomReadIOs + Δ SequentialReadIOs)
Percent Read Hits Random	Percentage (%) of random read I/O's that were de-staged from cache	%	100 x Δ RandomReadHitIOs / Δ RandomReadIOs
Percent Read Hits Seq	Percentage (%) of sequential read I/O's that were de-staged from cache	%	100 x Δ SequentialReadHitIOs / Δ SequentialReadIOs
Percent Reads	Percentage (%) of reads from cache	%	100 x (Δ RandomReadIOs + Δ SequentialReadIOs)/ (Δ RandomReadIOs + Δ RandomWriteIOs + Δ SequentialReadIOs + Δ SequentialWriteIOs)

Metric	Description	Units	Formulas
Percent Writes	Percentage (%) of writes from cache	%	100 x (Δ RandomWriteIOs + Δ SequentialWriteIOs)/ Δ RandomReadIOs + Δ RandomWriteIOs + Δ SequentialReadIOs + Δ SequentialWriteIOs)
Read Data Rate	Rate at which data is read from the array group by all hosts and includes transfers from the source array to the destination array	Bytes/s	(Δ RandomReadData + Δ SequentialReadData) / Δ StatisticTime
Read Data Rate Random	Rate at which random data is read from the array group by all hosts	Bytes/s	Δ RandomReadData / Δ StatisticTime
Read Data Rate Seq	Rate at which sequential data is read from the array group by all hosts and includes transfers from the source array to the destination array	Bytes/s	Δ SequentialReadData/ Δ StatisticTime
Read Hits	Cache read hits in requests per second	Req/s	(ΔRandomReadHitIOs + Δ SequentialReadHitIOs) / Δ StatisticTime
Read Response	Time required to complete a read I/O in seconds	Sec	Δ ReadResponseTimeCounter / (Δ RandomReadIOs + Δ SequentialReadIOs)
Reads	Number of read I/O's in requests per second	Req/s	(Δ RandomReadIOs + Δ SequentialReadIOs) / Δ StatisticTime
Total Data Rate	Rate in which data can be transmitted between devices for the selected array group	Req/s	(Δ RandomReadData + Δ RandomWriteData + Δ SequentialReadData + Δ SequentialWriteData) / Δ StatisticTime
Total I/O Rate	Total number of read or write operations taking place per second for the selected array group	Req/s	(Δ RandomReadIOs + Δ RandomWriteIOs + Δ SequentialReadIOs + Δ SequentialWriteIOs) / Δ StatisticTime

Metric	Description	Units	Formulas
Utilization	Percentage (%) of time that disks in the array group are busy	%	100 x Δ ActiveTime fields / Δ StatisticTime
Write Data Rate	Rate at which data is written to the array group by all hosts and includes transfers from the source array to the destination array	Bytes/s	(Δ RandomWriteData + Δ SequentialWriteData) / Δ StatisticTime
Write Data Rate Random	Rate at which random data is written to the array group by all hosts	Bytes/s	Δ RandomWriteData / Δ StatisticTime
Write Data Rate Seq	Rate at which sequential data is written to the array group by all hosts and includes transfers from the source array to the destination array	Bytes/s	Δ SequentialWriteData/ Δ StatisticTime
Write Response Time	Time required to complete a write I/O in seconds	Sec	Δ WriteResponseTimeCounter / (Δ RandomWriteIOs + Δ SequentialWriteIOs)
Writes	Number of write I/O's in requests per second	Req/s	(Δ RandomWriteIOs + Δ SequentialWriteIOs) / Δ StatisticTime

[1]Performance metrics in this table are not available for XP P9500 storage arrays discovered using Command View Advanced Edition (CVAE). For more information, see the Licensing chapter in the *Installation Guide*.

XP and HDS Front-end Controller CLPR Metrics

The following Cache Logical Partition (CLPR) metrics are available for the front-end controller.

Note: The MPB (Micro Processor Blade) controllers for XP P9500 storage arrays have a separate set of metrics to measure performance. These front-end controller CLPR metrics are therefore not used to measure the performance for XP P9500 arrays.

Metric	Description	Units	Formulas
Total Data Rate	Rate at which data can be transmitted between devices for the selected CLPR	Bytes\s	ΔTotallDataTransferred / Δ StatisticTime
Total I/O Rate	Total number of read or write operations taking place per second for the selected front-end controller CLPR	Req\s	Δ TotalIOs / Δ StatisticTime

XP and HDS Front-end Controller (CHA) Metrics

Front-end controllers route I/O to/from hosts to their disk cache slots. The front-end view shows all channel adapters (CHA) and their constituent ports. Selecting the CHA allows you to view the processor (MP) utilization.

Note: The MPB (Micro Processor Blade) controllers for XP P9500 storage arrays have a separate set of metrics to measure performance. This front-end controller (CHA) metric is therefore not used to measure performance for XP P9500 arrays.

Best Practices

Graph CHA MP Utilization and Port data rate simultaneously to determine if the port is the bottleneck. If MP utilization is less than 90%, and the port data rate is near its maximum, then you know that data is being de-staged to disk normally assuming the host(s) is running at its peak. If not, check the DKA and RAID Group utilization.

Controller CHA Metrics

The following performance metrics are available for the front-end controller.

Metric	Description	Units	Formula
Processor Utilization*	Processor utilization rate on the selected CHA controller	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter

*Metric not available for XP and HDS RAID700 arrays. XP and HDS RAID700 arrays utilize MPB controllers.

XP Back-end Controller (DKA) Metric

Back-end controllers route I/O from cache slots to the disk. The back-end view for XP arrays shows all the disk controller adapters (DKA). Selecting the DKA allows you to view processor utilization (MP).

The Hitachi Data Systems (HDS) Discovery process does not expose back-end controllers for HDS arrays. For this reason, there is no back-end controller (DKA) metric for HDS.

Note: The MPB (Micro Processor Blade) controllers for XP P9500 storage arrays have a separate set of metrics to measure performance. This back-end controller (DKA) metric is therefore not used to measure performance for XP P9500 arrays.

Best Practices

You can graph DKA MP Utilization and RAID Groups IOPS to determine if the controller is acting as a bottleneck. If MP utilization is around 90% and the RAID Group data rate is much lower than expected for the RAID level and disk type, there might be a background process such as replication that is over-utilizing the DKA. Typically, you want your back-up disks on a different RAID Group than your production disks, and the back-up disks should be connected to a different DKA.

Controller DKA Metrics

The following performance metrics are available for the XP back-end controller.

Metric	Description	Units	Formula
Processor Utilization*	Processor utilization rate on the selected DKA controller	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter

*Metric not available for XP and HDS RAID700 arrays. XP and HDS RAID700 arrays use MPB controllers.

XP and HDS MPB Controller Metrics

The following table describes the performance metrics collected for MPB (Micro Processor Blade) controllers for P9500/XP storage arrays (see footnote for exception). MBP processor statistics are obtained by HP Storage Essentials CIM extensions.

MPB Controller Metrics[1]

Metric	Description	Units	Formula
Back End Utilization	Processor utilization for the back-end processes (back-end activities for target I/O requests) as a percentage of total processor time.	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter
Mainframe External Initiator Utilization	Processor utilization for the mainframe external initiator processes as a percentage of total processor time.	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter
Mainframe Target Utilization	Processor utilization rate for the mainframe target processes (front-end activities for processing mainframe I/O requests) as a percentage of total processor time.	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter
Open External Initiator Utilization	Processor utilization rate for the open external initiator processes (external storage access activities) as a percentage of total processor time.	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter
Open Initiator Utilization	Processor utilization rate for the open initiator processes (continuous access replication activities) as a percentage of total processor time.	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter
Open Target Utilization	Processor utilization rate for open target processes (front-end activities) as a percentage of total processor time.	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter
Processor Utilization	Processor utilization rate on the selected MPB controller. This rate is the sum of the other seven metrics listed in this table.	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter
System Utilization	Processor utilization rate of the array system processes as a percentage of total processor time.	%	Δ BusyTimeCounter / Δ ElapsedTimeCounter

[1]The performance metrics in this table are not available for XP P9500 storage arrays discovered using Command View Advanced Edition (CVAE). For more information, see the Licensing chapter in the *Installation Guide*.

8 NetApp Performance Metrics

HP Storage Essentials Performance Pack Enterprise software for NetApp systems provides more than 80 performance metrics that you can use to monitor your NetApp storage devices. A complete list of performance metrics along with descriptions and derived formulas is provided in this section. See these topics for information:

- Monitoring NetApp Performance below
- NetApp System Performance Metrics on next page
- NetApp Raw Statistics on page 110

Monitoring NetApp Performance

HP Storage Essentials collects performance statistics derived from the NetApp devices in your storage environment and makes this data available to you in multiple ways for your monitoring and assessment activities. For example, you can use the Performance Manager to view data in real-time or chart historical data, and you can use Report Optimizer to generate reports and graphs for comparative analysis. Additionally, NetApp also provides information about monitoring NetApp devices for improved performance. You can find this information on the NetApp website at http://partners.netapp.com/go/techontap/matl/monitor-troubleshoot.html.

Using NetApp Performance Statistics to Identify Bottlenecks

HP Storage Essentials can help you identify bottlenecks and isolate potential problem areas before they affect your NetApp system. Two strong indicators of potential bottlenecks are throughput (I/O rates) and latency levels.

- Throughput is measured by a raw count of bytes or data packets transmitted and received along SAN connection paths; that is, along a storage system's Ethernet or target Fibre Channel interfaces. When a storage device performs at expected levels, without delays or queue backups, throughput statistics can be used as a measurement of baseline performance.
- Latency is the measured lapse in time between a data request send and its receipt (also called a round-trip loop). Latency can be measured for reads and writes: read and write hits and misses, reads and writes for a specific interface or protocol, and so forth. A low latency rate indicates optimum transmission of data.

Determine Latency Limits and Perform Baseline Analysis

Each application has acceptable limits for latency. Beyond these limits increasing degradation of performance becomes noticeable; user tasks take longer to execute and response times slow to unacceptable levels. Typically, this occurs over time as the number of users and applications increase, and heavier workloads are added to the existing storage system. Latency therefore is a strong indicator of how well your storage system handles workloads and where potential bottlenecks may develop over time.

However, what may be acceptable for one application or device, may be totally unacceptable for another. To determine the acceptable latency limits for your specific environment, establish a performance baseline that represents normal workloads and usage. Monitor your storage performance by comparing current, hourly, and daily performance data against your baseline statistics. If you expect heavier usage requirements at different times of the day, allow for this in establishing your performance baselines.

Add Performance Policies and Alerts

As an aid, set up performance policies with threshold alerts which advise you when latency limits are being exceeded. Like baselines, performance policies are a standard best practice for monitoring, evaluating, and verifying performance quality over time. See the *HP Storage Essentials Storage Performance Management Guide* for more information about setting up performance policies.

Understand NetApp Performance Metrics

You will find descriptions of the NetApp performance metrics available in HP Storage Essentials in NetApp System Performance Metrics below and NetApp Raw Statistics on page 110. These metrics are used in charting, reporting, policy creation, and real-time analysis of the performance of your NetApp devices.

NetApp System Performance Metrics

The following performance metrics are provided by HP Storage Essentials for NetApp storage systems, and include iSCSI, CIFS, DAFS, FCP, HTTP, and NFSv3 operations.

Metric	Description	Unit	Formula
Cache			
Buffer Cache Hits Count	Buffer Cache or system memory read cache hits per second. Use to determine if access latency is contributing to performance issues.	Req/s	Δ Buf_ Load_Cnt / Δ Time

Metric	Description	Unit	Formula
Buffer Cache Misses Count	Buffer cache miss count per second (rate). A cache miss is simply data that is not in the cache. The result is that the system must fetch the data from the disk. Use to determine PAM deployment requirements and configuration.	Req/s	Δ Buff_ Miss_Cnt / Δ Time
Inode Cache Hits Count	Number of hits for inodes read from disk that are cached and subsequently accessed. Use to determine cause of increase in file system performance	Req/s	Δ Inode_ Cache_ Hit / Δ Time
Inode Cache Misses Count	Number of inode cache misses per second (rate). A cache miss is simply data that is not in the cache. The result is that the system must fetch the data from the disk. The inode cache behaves in the same fashion. Use to determine if the inode cache needs to be increased.	Req/s	Δ Inode_ Cache_ Miss / Δ Time
Name Cache Hits	Number of name cache hits per second (rate). Use to determine frequency of name cache hits. The name cache improves file lookup in the file system.	Req/s	Δ Name_ Cache_ Hit / Δ Time
Name Cache Misses	Number of name cache misses per second (rate). A cache miss is simply data that is not in the cache. The result is that the system must fetch the data from the disk. The name cache behaves in the same fashion. Use to determine if the name cache needs to be increased.	Req/s	Δ Name_ Cache_ Miss / Δ Time
Common Internet File System (CIFS) I/O Operations			
CIFS Latency	Average latency for Common Internet File System (CIFS) operations in milliseconds.	ms	Δ CIFS_ Latency / Δ CIFS_ Latency_ Base
CIFS Operations	Number of Common Internet File System (CIFS) operations per second.	Req/s	Δ CIFS_ Ops / Δ Time
Direct Access File System (DAFS) I/O Operations			
DAFS Operations	Number of Direct Access File System (DAFS) operations per second.	Req/s	Δ DAFS_ Ops / Δ Time

Metric	Description	Unit	Formula
Fibre Channel Protocol (FCP) I/O Operations			
FCP Operations	Number of Fibre Channel Protocol (FCP) operations per second.	Req/s	Δ FCP_ Ops / Δ Time
FCP Read Data	FCP bytes read per second.	Bytes/s	Δ FCP_ Read_ Data / Δ Time
FCP Read Latency	Average latency for read operations observed over all LUNs in the system accessed over FCP in milliseconds.	ms	Δ FCP_ Read _ Latency / Δ FCP_ Read_ Ops
FCP Read Operations	Total number of read operations per second observed over all the LUNS in the system accessed over FCP.	Req/s	Δ FCP_ Read_ Ops / Δ Time
	FCP bytes written per second.	Bytes/s	Δ FCP_ Write_ Data / Δ Time
FCP Write Latency	Average latency for write operations observed over all LUNs in the system accessed over FCP in milliseconds.	ms	Δ FCP_ Write _ Latency / Δ FCP_ Write_ Ops
FCP Write Operations	Total number of write operations per second observed over all the LUNs in the system accessed over FCP.	Req/s	Δ FCP_ Write_ Ops / Δ Time
Hypertext Transfer Protocol (HTTP) I/O Operations			
HTTP Operations	Number of Hypertext Transfer Protocol (HTTP) operations per second.	Req/s	Δ HTTP_ Ops / Δ Time
Internet Small Computer System Interface (iSCSI) I/O Operations			

Metric	Description	Unit	Formula
ISCSI Operations	Number of Internet Small Computer System Interface (iSCSI) operations per second.	Req/s	Δ ISCSI_ Ops / Δ Time
ISCSI Read Data	iSCSI bytes read per second.	Bytes/s	Δ ISCSI_ Read_ Data / Δ Time
ISCSI Read Latency	Average latency of read operations observed over all LUNs in the system accessed over iSCSI in milliseconds.	ms	Δ ISCSI_ Read _ Latency / Δ ISCI_ Read_ Ops
ISCSI Read Operations	Total number of read operations per second observed over all the LUNs in the system accessed over iSCSI.	Req/s	Δ ISCSI_ Read_ Ops / Δ Time
ISCSI Write Data	iSCSI bytes written per second.	Bytes/s	Δ ISCSI_ Write_ Data / Δ Time
ISCSI Write Latency	Average latency of write operations observed over all LUNs in the system accessed over iSCSI in milliseconds.	ms	Δ ISCSI_ Write_ Latency / Δ ISCI_ Write_ Ops
ISCSI Write Operations	Total number of write operations per second observed overall the LUNs in the system accessed over iSCSI.	Req/s	Δ ISCSI_ Write_ Ops / Δ Time
Network File System (NFS) Operations			
NFSV3 Average Operations Latency	Average latency of the NFS v3 operations in milliseconds.	ms	Δ NFSv3_ Avg_Op _ Latency / Δ NFSv3_ Avg_Op_ Latency_ Base

Metric	Description	Unit	Formula
NFS Operations	Number of Network File System (NFS) operations per second.	Req/s	Δ NFS_ Ops / Δ Time
NFSV3 Read Latency	Average latency for NFS v3 read operations in milliseconds.	ms	Δ NFSv3_ Read _ Latency / Δ NFSv3_ Avg_ Read_ Latency_ Base
NFSV3 Write Latency	Average latency for NFS v3 write operations in milliseconds.	ms	Δ NFSv3_ Write _ Latency / Δ NFSv3_ Avg_ Write_ Latency_ Base

NetApp Host Processor (CPU) Performance Metrics

The following performance metrics are provided for NetApp host processors (CPU).

NetApp does not expose memory utilization statistics because all memory that is not used for initialization is allocated to the file system for caching. In other words, all memory is used all of the time. How it is allocated is what really matters. Processor utilization and cache hit ratio are better indicators of how utilized a filer is. If the cache hit ratio is low and CPU utilization is high, the filer might be reaching maximum utilization.

Metric	Description	Unit	Formula
Processor Utilization	Total CPU utilization (%) by all the processes running on the filer. Indicates the percentage (%) of time that the processor is active. A completely idle processor shows 0%. A processor saturated with activity shows 100%. Use to identify CPU bottlenecks.	%	100 x (Δ Processor_ Busy / Δ Processor_ Elapsed_ Time)

NetApp Aggregate Performance Metrics

The following metric is provided for measuring NetApp aggregate transfer rates.

Metric	Description	Unit	Formula
Total Transfers	Total number of transfers per second serviced by the aggregate.	Req/s	Δ Total_Transfers / Δ Time
User Read Block	Number of blocks read per second on the aggregate.	Blocks/s	Δ User_Read_ Blocks / Δ Time
User Reads	Number of user reads per second to the aggregate.	Req/s	Δ User_Reads / Δ Time
User Write Blocks	Number of blocks written per second to the aggregate.	Blocks/s	Δ User_Write_ Blocks / Δ Time
User Writes	Number of user writes per second to the aggregate.	Req/s	Δ User_Writes / Δ Time

NetApp Front-end FC Port Performance Metrics

The following front-end Fibre Channel (FC) port metrics measure performance state using HP Storage Essentials.

Metric	Description	Unit	Formula
Other Operations	Number of other operations per second.	Req/s	Δ Other_Ops / Δ Time
Queue Full Responses	SCSI queue full responses per second.	Req/s	Δ Queue_Full / Δ Time
Read Data	Number of bytes read from the filer per second.	Bytes/s	Δ Read_Data / Δ Time
Read Operations	Number of read operations per second.	Req/s	Δ Read_Ops / Δ Time
Write Data	Number of bytes written to the filer per second.	Bytes/s	Δ Write_Data / Δ Time
Write Operations	Number of write operations per second.	Req/s	Δ Write_Ops / Δ Time

NetApp IP Port Performance Metrics

The following metrics are available to measure NetApp IP port performance.

Metric	Description	Unit	Formula
Bytes Received	Inbound traffic in megabytes per second through the filer network interface controller (NIC). Use to assess network traffic for load balancing, multi-path optimization, and network performance.	MB/s	(Δ Recv_ Data x 1024) / Δ Time
Bytes Transmitted	Outbound traffic in megabytes per second through the filer network interface controller (NIC). Use to assess network traffic for load balancing, multi-path optimization, and network performance.	MB/s	(Δ Send_ Data x 1024) / Δ Time
Packets Received	Inbound traffic in packets per second through the filer network interface controller (NIC). Network packets contain data headers, address source and destination, payload and CRC fields. Use to measure network traffic for load balancing, multi-path optimization, and network performance. Contrary to bytes received testing, packet testing is actually a better test because inbound packets have either arrived or not. Byte testing does not indicate whether or not a packet transmission completed.	Pkt/s	Δ Recv_ Packets / Δ Time
Packets Transmitted	Outbound traffic in packets per second through the filer network interface controller (NIC). Network packets contain data headers, address source and destination, payload and CRC fields. Use to measure network traffic for load balancing, multi-pathing optimization and network performance. Contrary to bytes transmitted testing, packet testing is actually a better test because outbound packets have been either sent or not. Byte testing does not indicate whether or not a packet transmission completed.	Pkt/s	Δ Send_ Packets / Δ Time
Receive Errors	Errors per second while receiving packets.	Req/s	Δ Recv_ Errors / Δ Time
Send Errors	Errors per second while sending packets.	Req/s	Δ Send_ Errors / Δ Time

NetApp Logical Drive Performance Metrics

The following NetApp logical drive performance metrics are available. The metrics for NetApp file system and volume statistics are combined.

Metric	Description	Unit	Formula
Average Latency	Average latency in milliseconds for all operations on the volume.	ms	Δ Avg_ Latency / Δ Total_ Ops
Other Latency	Average latency time for other writes to the volume in milliseconds.	ms	Δ Other_ Latency / Δ Other_ Ops
Other Operations	Number of other operations per second to the volume.	Req/s	Δ Other_ Ops / Δ Time
Read Data	Bytes read per second from the volume.	Bytes/s	Δ Read_ Data / Δ Time
Read Latency	Average latency time for reads to the volume in milliseconds.	ms	Δ Read_ Latency / Δ Read_ Ops
Read Operations	Number of reads per second to the volume.	Req/s	Δ Read_ Ops / Δ Time
Reserved Inodes	Provides the count of reserved inodes in a file system. Use to count the reserved inodes. The first 10 inodes on a file system are special inodes.Inodes 7-10 are reserved and usually not used.	counter	–
Total Inodes	Total number of inodes. Inodes are file system data structures or metadata used to store basic file data like ownership and file permissions. Use to view the inode limit and determine if more are needed.	counter	–
Total Operations	Total number of operations per second serviced by the volume.	Req/s	Δ Total_ Ops / Δ Time

Metric	Description	Unit	Formula
Used Inodes	Total number of inodes that are currently used. Use to show the number of inodes in use. Use to alert the admin when inode utilization is approaching total inodes available.	counter	–
Write Data	Bytes written per second to the volume.	Bytes/s	Δ Write_Data / Δ Time
Write Latency	Average latency time for writes to the volume in milliseconds.	ms	Δ Write_Latency / Δ Write_Ops
Write Operations	Number of writes per second to the volume.	Req/s	Δ Write_Ops / Δ Time

NetApp LUN Performance Metrics

The following metrics measure NetApp LUN performance.

Metric	Description	Unit	Formula
Average Latency	Average latency in milliseconds for all operations on the LUN	ms	Δ Avg_Latency / Δ Total_Ops
Other Operations	Number of other operations per second	Req/s	Δ Other_Ops / Δ Time
Queue Full Responses	Number of queue full responses per second	Req/s	Δ Queue_Full / Δ Time
Read Data	Number of bytes read per second	Bytes/s	Δ Read_Data / Δ Time
Read Operations	Number of read operations per second	Req/s	Δ Read_Ops / Δ Time
Total Operations	Total number of operations on the LUN per second	Req/s	Δ Total_Ops / Δ Time
Write Data	Number of bytes written per second	Bytes/s	Δ Write_Data / Δ Time
Write Operations	Number of write operations per second	Req/s	Δ Write_Ops / Δ Time

NetApp Disk Drive Performance Metrics

The following performance measurements are available for NetApp disk drives.

Metric	Description	Unit	Formula
Disk Busy	Percentage of time there was at least one outstanding request to the disk	%	100 x (Δ Disk_ Busy / Δ Disk_ Busy_Base)
Total Transfers	Total number of disk operations involving data transfer initiated per second	Req/s	Δ Total_ Transfers / Δ Time
User Read Blocks	Number of blocks transferred for user read operations per second	Blocks/s	Δ User_Read_ Blocks / Δ Time
User Read Latency	Average latency per block in milliseconds for user read operations	ms	Δ User_Read_ Latency / Δ User_Read_ Blocks
User Reads	Number of disk read operations initiated each second for retrieving data or metadata associated with user request	Req/s	Δ User_Reads / Δ Time
User Write Blocks	Number of blocks transferred for user write operations per second	Blocks/s	Δ User_Write_ Blocks / Δ Time
User Write Latency	Average latency per block in milliseconds for user write operations	ms	Δ User_Write_ Latency / Δ User_Write_ Blocks
User Writes	Number of disk write operations initiated each second for retrieving data or metadata associated with user requests	Req/s	Δ User_Writes / Δ Time

NetApp Qtree Performance Metrics

The following performance metrics are provided for the NetApp QTree.

Metric	Description	Unit	Formula
CIFS Operations	Number of Common Internet File System (CIFS) operations per second to the qtree	Req/s	Δ CIFS_ Ops / Δ Time
Internal Operations	Number of internal operations generated by activities such as snapmirror and backup per second to the qtree	Req/s	Δ Internal_ Ops / Δ Time
NFS Operations	Number of NFS operations per second to the qtree	Req/s	Δ NFS_ Ops / Δ Time

NetApp Raw Statistics

The following raw values calculate the performance metric data provided in the Performance Manager for NetApp system devices. All raw metrics are counters derived from the NetApp device.

For a list of the Performance Manager metrics that use these raw statistics, see the NetApp System Performance Metrics on page 100.

NetApp System Raw Statistics

The following values are derived from the NetApp device and are stored in the NAS_SYSTEM_ STATS database table. This database contains the iSCSI, CIFS, FCP, NFSv3 and general system statistics.

Raw Statistic	Description	Unit
buf_load_cnt	Number of buffer cache or system memory read cache hits.	Hit count
buff_miss_cnt	Number of buffer cache misses.	Miss count
cifs_latency	Average latency for Common Internet File System (CIFS) operations.	Time count in milliseconds
cifs_latency_ base	Total observed CIFS operations to be used as a base counter for CIFS average latency calculation.	Time count in milliseconds
cifs_ops	Number of Common Internet File System (CIFS) operations.	Request count
dafs_ops	Number of Direct Access File System (DAFS) operations.	Request count
fcp_ops	Number of Fibre Channel Protocol (FCP) operations.	Request count

Raw Statistic	Description	Unit
fcp_read_data	Number of FCP bytes read.	Byte count
fcp_read_latency	Average latency for read operations observed over all LUNs in the system accessed over FCP.	Time count in milliseconds
fcp_read_ops	Total number of read operations observed over all the LUNS in the system accessed over FCP.	Request count
fcp_write_data	Number of FCP bytes written.	Byte count
fcp_write_latency	Average latency for write operations observed over all LUNs in the system accessed over FCP.	Time count in milliseconds
fcp_write_ops	Total number of write operations observed over all the LUNs in the system accessed over FCP.	Request count
http_ops	Number of Hypertext Transfer Protocol (HTTP) operations.	Request count
inode_cache_hit	Number of hits for inodes read from disk that are cached.	Hit count
inode_cache_miss	Number of inode cache misses.	Miss count
iscsi_ops	Number of Internet Small Computer System Interface (iSCSI) operations.	Request count
iscsi_read_data	iSCSI bytes read.	Byte count
iscsi_read_latency	Average latency of read operations observed over all LUNs in the system accessed over iSCSI.	Time count in milliseconds
iscsi_read_ops	Total number of read operations per second observed over all the LUNs in the system accessed over iSCSI.	Request count
iscsi_write_data	Number of iSCSI bytes written.	Byte count
iscsi_write_latency	Average latency of write operations observed over all LUNs in the system accessed over iSCSI.	Time count in milliseconds
iscsi_write_ops	Total number of write operations observed on all LUNs in the system accessed over iSCSI.	Request count
name_cache_hit	Number of name cache hits.	Hit count

Raw Statistic	Description	Unit
name_cache_ miss	Number of name cache misses.	Miss count
nfsv3_avg_ op_latency	Average latency of the NFS v3 operations.	Time count in milliseconds
nfsv3_avg_ op_latency_ base	Array of select NFS v3 operation counts for latency calculation.	Time count in milliseconds
nfs_ops	Number of Network File System (NFS) operations.	Request count
nfsv3_read_ latency	Average latency for NFS v3 read operations.	Time count in milliseconds
nfsv3_write_ latency	Average latency for NFS v3 write operations.	Time count in milliseconds

NetApp Aggregate Raw Statistics

The following values are derived from the NetApp device and are stored in the NAS_ AGGREGATE_STATS database table.

Raw Statistic	Description	Unit
total_transfers	Number of transfers.	Request count
user_read_blocks	Number of blocks read.	Data block count
user_reads	Number of user reads.	Request count
user_write_blocks	Number of blocks written.	Data block count
user_writes	Number of user writes.	Request count

NetApp Logical Drive Raw Statistics

The following values are derived from the NetApp device and are stored in the NAS_ FILESYSTEM_STATS database table. This database contains the volume and file system statistics.

Raw Statistic	Description	Unit
avg_latency	Average latency for all operations on the volume.	Time count in milliseconds

Raw Statistic	Description	Unit
other_latency	Latency time for other writes to the volume.	Time count in milliseconds
other_ops	Number of other operations to the volume.	Request count
read_data	Number of bytes read from the volume.	Byte count
read_latency	Average latency time for reads to the volume.	Time count in milliseconds
read_ops	Number of read operations to the volume.	Request count
total_ops	Total number of operations serviced by the volume.	Request count
write_data	Number of bytes written to the volume.	Byte count
write_latency	Average latency time for writes to the volume.	Time count in milliseconds
write_ops	Number of write operations to the volume.	Request count
wv_fsinfo_inos_reserve	Reserved inodes in the volume.	Inode count
wv_fsinfo_inos_total	Total number of inodes in the volume.	Inode count
wv_fsinfo_inos_used	Used inodes in the volume.	Inode count

NetApp Processor (CPU) Raw Statistics

The following values are derived from the NetApp device and are stored in the HOSTCPUSTAT database table.

Metric	Description	Unit
processor_ busy	Time CPU is being utilized. Uses the wall-clock time since boot and is used for calculating processor utilization.	Time count in seconds
processor_ elapsed_ time	Time CPU is running, including busy and idle periods.	Time count in seconds

NetApp Front-end Port Raw Statistics

The following values are derived from the NetApp device and are stored in the NAS_FCPORT_ STATS database table.

Raw Statistic	Description	Unit
other_ops	Number of other operations to the volume.	Request count

Raw Statistic	Description	Unit
read_data	Number of bytes read from the filer.	Byte count
read_ops	Number of read operations to the volume.	Request count
write_data	Number of bytes written to the filer.	Byte count
write_ops	Number of write operations to the volume.	Request count
queue_full	Number of responses that SCSI queue is full.	Response count

NetApp IP Port Raw Statistics

The values for the following statistics are derived from the NetApp device and are stored in the IP_PORTSTATS database table.

Raw Statistic	Description	Unit
recv_ data	Amount of inbound data in bytes received through the filer network interface controller (NIC).	Byte count
recv_ errors	Number of errors which occurred while receiving packets.	Error count
recv_ packets	Number of inbound packets received through the filer network interface controller (NIC).	Packet count
send_ data	Amount of outbound data in bytes sent through the filer network interface controller (NIC).	Byte count
send_ errors	Number of errors which occurred while sending packets.	Error count
send_ packets	Number of outbound packets sent through the filer network interface controller (NIC).	Packet count

NetApp LUN Raw Statistics

The following values for statistics are derived from the NetApp device and are stored in the NAS_ LUN_STATS database table.

Raw Statistic	Description	Unit
avg_ latency	Average latency time count for all operations on the LUN.	Time count in milliseconds
other_ops	Number of other operations.	Request count

Raw Statistic	Description	Unit
queue_full	Number of queue full responses.	Response count
read_data	Number of bytes read.	Byte count
read_ops	Number of read operations from the LUN.	Request count
total_ops	Number of total operations on the LUN.	Request count
write_data	Number of bytes written to the LUN.	Byte count
write_ops	Number of write operations to the LUN.	Request count

NetApp Disk Drive Raw Statistics

The following values for statistics are derived from the NetApp device and are stored in the NAS_DISKDRIVE_STATS database table.

Raw Statistic	Description	Unit
disk_busy	Time count for when there was at least one outstanding request to the disk.	Time count
disk_busy_base	Time base for disk_busy calculation.	Time count
total_transfers	Total number of disk operations involving data transfers.	Request count
user_read_blocks	Number of blocks transferred for user read operations.	Block count
user_read_latency	Latency for user read operations.	Time count in microseconds
user_reads	Number of disk read operations associated with a user request.	Request count
user_write_blocks	Number of blocks transferred for user write operations.	Block count
user_write_latency	Latency for user write operations.	Time count in microseconds
user_writes	Number of disk write operations associated with user requests.	Request count

NetApp QTree Raw Statistics

The following values are derived from the NetApp device and stored in the NAS_QTREE_STATS database table.

Raw Statistic	Description	Unit
cifs_ops	Number of Common Internet File System (CIFS) operations to the qtree.	Request count
Internal_ops	Number of internal operations generated by activities (such as snapmirror and backup) to the qtree.	Request count
nfs_ops	Number of NFS operations to the qtree.	Request count