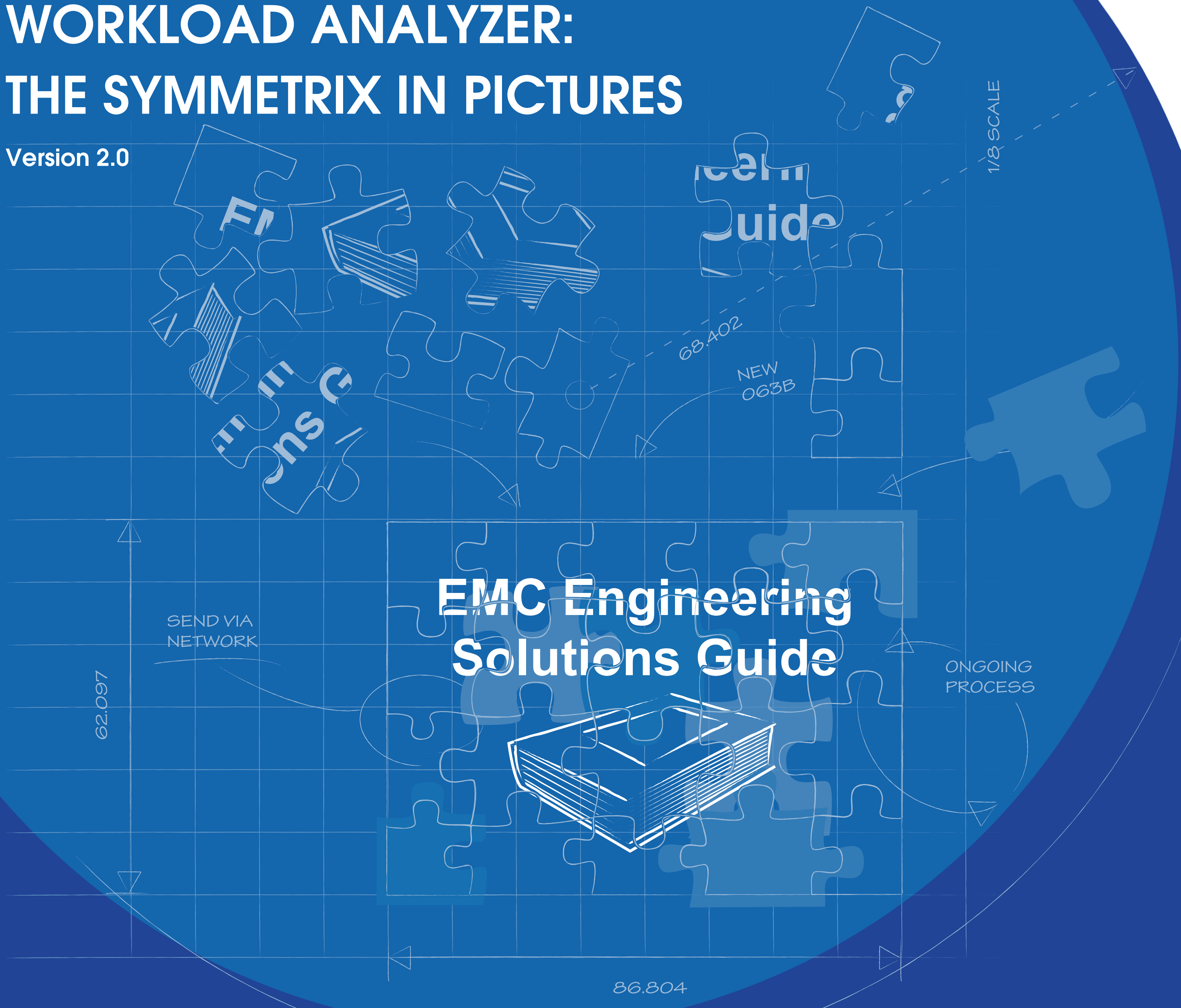
EMC²
where information lives

# EMC CONTROLCENTER WORKLOAD ANALYZER: THE SYMMETRIX IN PICTURES

Version 2.0



Author

Bill Zahavi

Corporate Information

Corporate Headquarters:
Hopkinton, MA 01748-9103
(800) 424-EMC2
http://www.EMC.com

Part number C1087

# Contents

# Preface

The Workload Analyzer (WLA) is a collection of software divided into independent components that run on the hardware in the storage area network (SAN). There are components that run on the application hosts, management hosts, the storage arrays, and the fibre switches-directors. Together this software is responsible for collecting data that relates to SAN performance and management.

This document describes the WLA and how to use it. It gives many examples that are important for anyone who intends to use WLA. This tool is useful for the novice and master alike. This is an introduction to the product and leads you through many examples of situations that occur in everyday operations.

The WLA is often labeled an 'expert' tool requiring navigational skill of the tool and an understanding of performance. Performance information is easy to access in order to reduce the need for experts to run the tool. Factory views and automated reports are included in the WLA to reduce the time needed to be productive with the tool.

## Purpose

Application performance is dependent on multiple SAN layers comprising both hardware and software components. The SAN layers encompass the storage arrays, switches, hosts and multiple host software layers such as databases, volume managers, file systems, and operating systems. To demonstrate the features and benefits of the Workload Analyzer, this document focuses on the analysis of the storage array layer of the SAN. More specifically, the document uses the Symmetrix® array in its description of a storage array as well as in the process of conducting performance analysis.

This is a tool for performance analysts and storage administrators alike. This document orients them to this tool in a productive way. It highlights the key features of all components. It explains the Symmetrix Storage system in an insightful and abbreviated manner.

The document includes many examples that illustrate how to view performance data and how to use several techniques for interpreting the data.

## Structure

This paper is divided into three major sections:

Chapter 1, *Effective Use of the Workload Analyzer*, describes the main components of the Workload Analyzer.

Chapter 2, *The Symmetrix Architecture: A Brief Overview*, discusses the key areas of the Symmetrix array that affect performance.

Chapter 3, *Analyzing Symmetrix Performance*, uses a series of experiments to show how performance is analyzed from data collected at both the Symmetrix and a connected host.

The focus of the analysis is on measured response times as measurements of system service quality.

## Summary

The EMC ControlCenter™ Workload Analyzer provides the analyst with the tools to understand the environment monitored and managed with EMC ControlCenter. This document describes the key features of the Workload Analyzer and the Symmetrix storage array.

Several examples are used to illustrate how to navigate around the performance metrics in order to evaluate storage-system performance with the available data. Other aspects to performance management, such as workload characterization, sizing, capacity planning, and tuning, are offshoots of performance analysis that can only be carried out successfully when the analyst is first familiar with the data and the data analysis tools. It is the intent of this document to create this familiarity.

This document covers ControlCenter Version 5.1.2.

1

# Effective Use of the Workload Analyzer

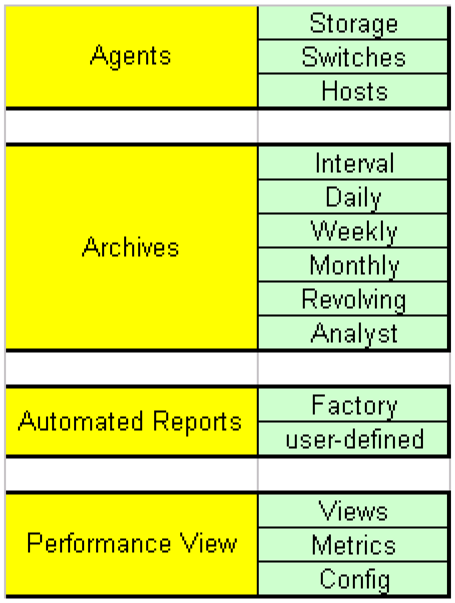
This chapter reviews these topics:

## 1.1 Overview

Four major components make up the Workload Analyzer (WLA):

- Data Collection Agents
- Archives
- Automated Reports
- Performance View

Each component contains unique features and functions. The agents collect and aggregate data for the storage arrays, switches, and hosts. The archives are where the performance data is kept. The archive data is organized according to intervals that are explained in detail later in this document. Automated reports provide an easy way to create standard reports that can be executed quickly. Using Performance View is an easy way to query the WLA data to answer performance questions.



| | |
|---|---|
| Agents | Storage |
| | Switches |
| | Hosts |
| Archives | Interval |
| | Daily |
| | Weekly |
| | Monthly |
| | Revolving |
| | Analyst |
| Automated Reports | Factory |
| | user-defined |
| Performance View | Views |
| | Metrics |
| | Config |

**Figure 1-1 Workload Analyzer Components: Features and Functions**

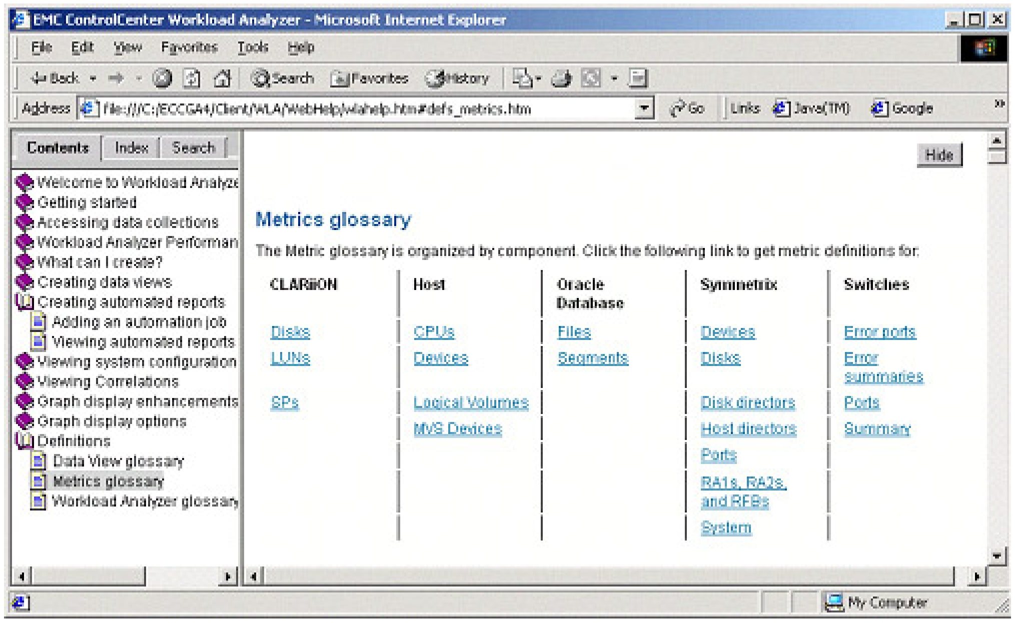
The purpose of this chapter is to highlight the function of each of the components and to familiarize you with some of the views associated with each component. This chapter is meant to provide a broad-brush view of the product and should lead you to the user's guide as well as to the product itself for further exploration.

## 1.2 Agents

The EMC ControlCenter agents serve as the data collectors for the Workload Analyzer. The ControlCenter agents perform many active and passive functions relative to SAN management and, as such, they cover a broad range of heterogeneous devices. Not all of the devices currently provide ControlCenter with performance data. Agents collecting performance data exist for the following entities:

| | | | |
|---|---|---|---|
| **Storage Systems** | Symmetrix | | |
| | CLARiiON | | |
| **Switches** | All used by EMC storage systems | | |
| | **OS** | **Volume Mgr.** | **DB** |
| **HOSTS** | Sun Solaris | VERITAS | Oracle |
| | HP-UX | VERITAS | Oracle |
| | Windows | | Oracle |
| | AIX | | Oracle |
| | MVS | | |

A detailed description of the metrics for each of the data providers is best viewed from the online Help facility of Performance View. In addition to the standard feature descriptions, WLA provides a complete metrics glossary with easy navigation.


EMC ControlCenter Workload Analyzer - Microsoft Internet Explorer

Address: file:///C:/ECCGA4/Client/WLA/WebHelp/wlahelp.htm#defs_metrics.htm

Contents | Index | Search

- Welcome to Workload Analyze
- Getting started
- Accessing data collections
- Workload Analyzer Performan
- What can I create?
- Creating data views
- Creating automated reports
  - Adding an automation job
  - Viewing automated reports
- Viewing system configuration
- Viewing Correlations
- Graph display enhancements
- Graph display options
- Definitions
  - Data View glossary
  - Metrics glossary
  - Workload Analyzer glossary

Metrics glossary

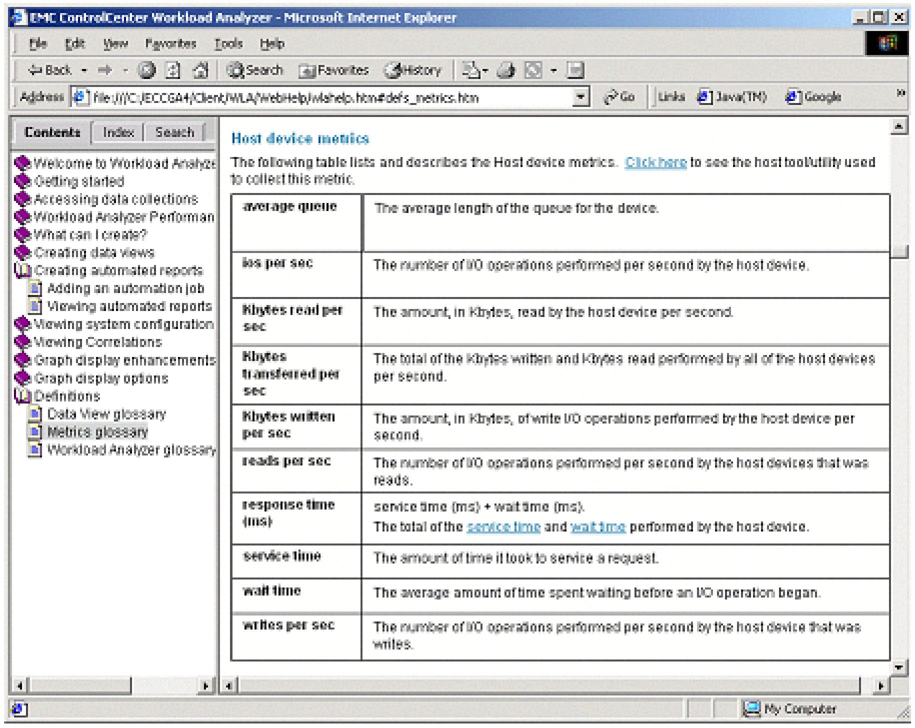
The Metric glossary is organized by component. Click the following link to get metric definitions for:

| CLARiiON | Host | Oracle Database | Symmetrix | Switches |
|---|---|---|---|---|
| Disks | CPUs | Files | Devices | Error ports |
| LUNs | Devices | Segments | Disks | Error summaries |
| SPs | Logical Volumes | | Disk directors | Ports |
| | MVS Devices | | Host directors | Summary |
| | | | Ports | |
| | | | RA1s, RA2s, and RFBs | |
| | | | System | |

File Edit View Favorites Tools Help
Back Search Favorites History
Go Links Java(TM) Google
Hide
My Computer

Figure 1-2 **Help Facility of Performance View**

By selecting a category of metrics, you are provided with both the metric name as well as a detailed description of that metric.


EMC ControlCenter Workload Analyzer - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address file:///C:/ECCGA4/Client/WLA/WebHelp/wlahelp.htm#defs_metrics.htm
Go Links Java(TM) Google
Contents Index Search
Welcome to Workload Analyze
Getting started
Accessing data collections
Workload Analyzer Performan
What can I create?
Creating data views
Creating automated reports
Adding an automation job
Viewing automated reports
Viewing system configuration
Viewing Correlations
Graph display enhancements
Graph display options
Definitions
Data View glossary
Metrics glossary
Workload Analyzer glossary
My Computer

Host device metrics

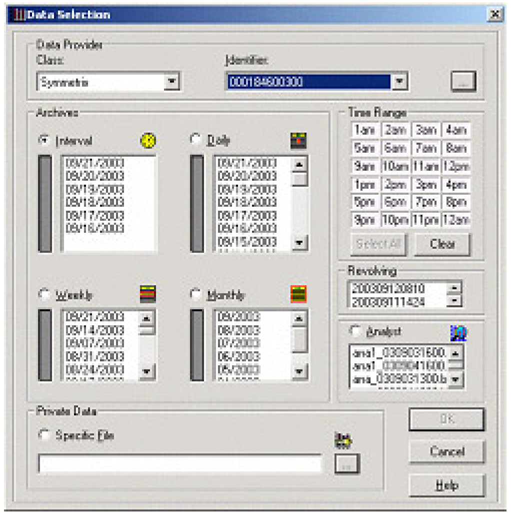
The following table lists and describes the Host device metrics. Click here to see the host tool/utility used to collect this metric.

| | |
|---|---|
| **average queue** | The average length of the queue for the device. |
| **ios per sec** | The number of I/O operations performed per second by the host device. |
| **Kbytes read per sec** | The amount, in Kbytes, read by the host device per second. |
| **Kbytes transferred per sec** | The total of the Kbytes written and Kbytes read performed by all of the host devices per second. |
| **Kbytes written per sec** | The amount, in Kbytes, of write I/O operations performed by the host device per second. |
| **reads per sec** | The number of I/O operations performed per second by the host devices that was reads. |
| **response time (ms)** | service time (ms) + wait time (ms).<br>The total of the service time and wait time performed by the host device. |
| **service time** | The amount of time it took to service a request. |
| **wait time** | The average amount of time spent waiting before an I/O operation began. |
| **writes per sec** | The number of I/O operations performed per second by the host device that was writes. |

**Figure 1-3 Displaying Metric Name and a Detailed Description of Metric**

## 1.3 The Archives

The archives are the sets of data files where historical performance data is maintained. Access to the performance archives is via the Workload Analyzer Performance View component. The best way to describe the archives is in terms of the selection window that appears when you open up PerformanceView.



Class: Symmetrix
Identifier: 000184600300

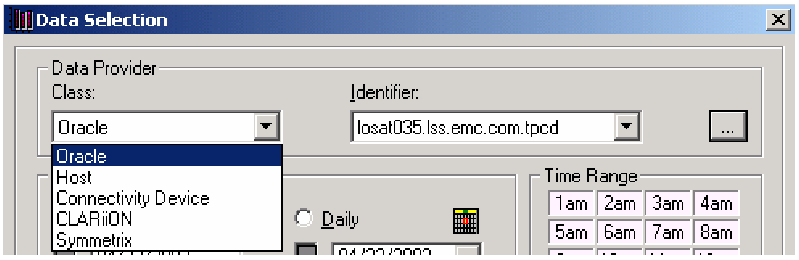
Interval: 09/21/2003, 09/20/2003, 09/19/2003, 09/18/2003, 09/17/2003, 09/16/2003
Daily: 09/21/2003, 09/20/2003, 09/19/2003, 09/18/2003, 09/17/2003, 09/16/2003, 09/15/2003
Weekly: 09/21/2003, 09/14/2003, 09/07/2003, 08/31/2003, 08/24/2003
Monthly: 09/2003, 08/2003, 07/2003, 06/2003, 05/2003

| | | | |
|---|---|---|---|
| 1am | 2am | 3am | 4am |
| 5am | 6am | 7am | 8am |
| 9am | 10am | 11am | 12pm |
| 1pm | 2pm | 3pm | 4pm |
| 5pm | 6pm | 7pm | 8pm |
| 9pm | 10pm | 11pm | 12am |

Revolving: 200309120810, 200309111424
Analyst: ana1_0309031600, ana1_0309041600, ana_0309031300.b

Figure 1-4 **Workload Analyzer Performance View**

If data collection agents are installed and turned on, you can see the available data collection categories in the Class pull-down menu:



Class: Oracle
Identifier: losat035.lss.emc.com.tpcd

- Oracle
- Host
- Connectivity Device
- CLARiiON
- Symmetrix

Figure 1-5 **Data Collection Categories in the Class Pull-down Menu**

Having selected a Data Provider class, you can then select from the available data provider instances. In the case of the following example, the Symmetrix instances are listed using the Symmetrix serial numbers.

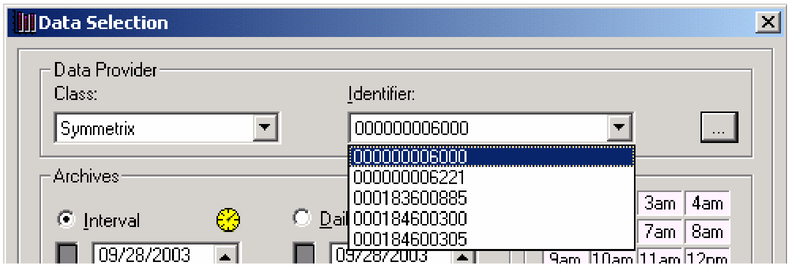


Figure 1-6 **Data Provider Class: Symmetrix**

Each data provider with an active agent has three types of data collection policies:

- Interval collection
- Revolving collection
- Analyst collection

## 1.3.1 Interval Collection

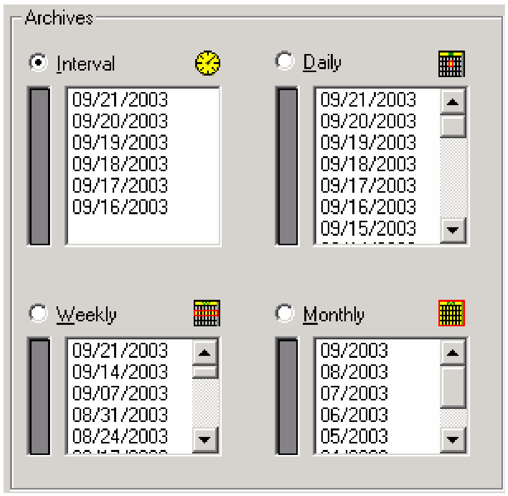
This policy establishes the time interval for which continuous data collection is done. It is initially set to 15 minutes (default), but you can change this value. Smaller time intervals will generate larger data files.

Interval data is stored in the archives on a daily basis and is subject to a retention policy initially set by default to 7 days. The time interval and the number of data collection entities for which data is collected from the data provider define the size of the Interval data file. For example, on a Symmetrix array, the primary factor for the size of the file is the number of logical devices configured.

The interval data is the initial step for the historical archives. The archiving engine of the Workload Analyzer statistically compresses the data into daily, weekly, and monthly files. This occurs as follows:

- **A** — For each hour of the day, the interval data points for that hour are gathered and the average value with the maximum value are retained for that hour in the daily file.
- **B** — For each hour of the day and for each day of the week, the average value and the maximum value are retained for that hour in the weekly file.

  For example, the weekly file will look very similar to a daily file when graphed because it will have 24 hourly points. However, each hour is the average value for the whole week.
- **C** — For each hour of the day and for each day of the month, the average value and maximum value are retained for that hour in the monthly file.



Interval: 09/21/2003, 09/20/2003, 09/19/2003, 09/18/2003, 09/17/2003, 09/16/2003

Daily: 09/21/2003, 09/20/2003, 09/19/2003, 09/18/2003, 09/17/2003, 09/16/2003, 09/15/2003

Weekly: 09/21/2003, 09/14/2003, 09/07/2003, 08/31/2003, 08/24/2003

Monthly: 09/2003, 08/2003, 07/2003, 06/2003, 05/2003

**Figure 1-7 Interval Data**

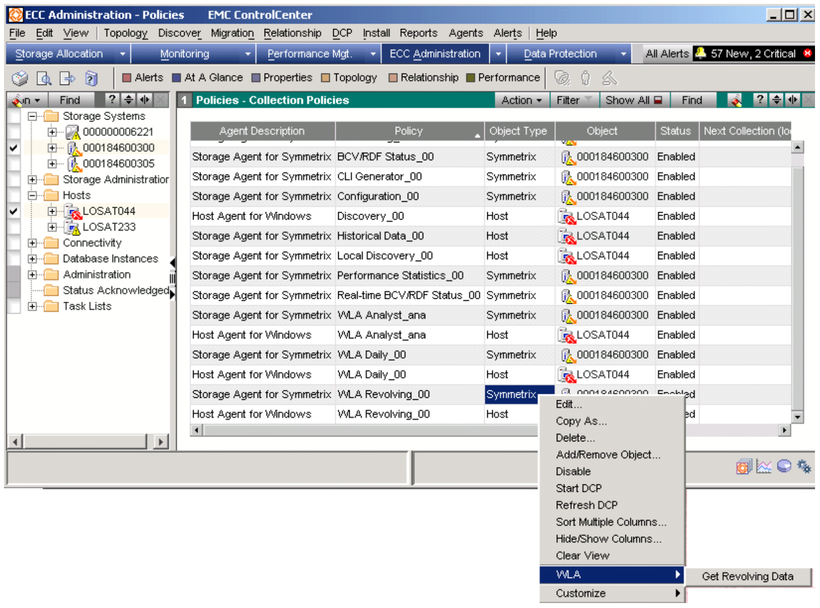
## 1.3.2 Revolving Collection

The revolving collection acts like a flight recorder.

You establish the collection interval frequency (default = 2 minutes) and the loop duration (default = 2 hours) using the policy setting. By doing this, at any time you can request to download the last two hours of WLA data for the selected data provider.

This feature is best used to analyze immediate performance problems because it provides an immediate view of the last 2 hours (or whatever duration the user previously set by policy).

Revolving collections run continuously on the system where the agent resides that manages the specific data provider. The data is maintained locally near the agent and is not transferred anywhere unless specifically requested. You request the revolving data by using the ControlCenter Console in the Policy view and by right-clicking to bring up the menu that shows the command to get the revolving WLA data.


ECC Administration - Policies EMC ControlCenter
File Edit View Topology Discover Migration Relationship DCP Install Reports Agents Alerts Help
Storage Allocation Monitoring Performance Mgt. ECC Administration Data Protection All Alerts 57 New, 2 Critical
Alerts At A Glance Properties Topology Relationship Performance
Find
Storage Systems
000000006221
000184600300
000184600305
Storage Administratior
Hosts
LOSAT044
LOSAT233
Connectivity
Database Instances
Administration
Status Acknowledged
Task Lists
1 Policies - Collection Policies
Action Filter Show All Find
Edit...
Copy As...
Delete...
Add/Remove Object...
Disable
Start DCP
Refresh DCP
Sort Multiple Columns...
Hide/Show Columns...
Clear View
WLA
Get Revolving Data
Customize

| Agent Description | Policy | Object Type | Object | Status | Next Collection (lo |
|---|---|---|---|---|---|
| Storage Agent for Symmetrix | BCV/RDF Status_00 | Symmetrix | 000184600300 | Enabled | |
| Storage Agent for Symmetrix | CLI Generator_00 | Symmetrix | 000184600300 | Enabled | |
| Storage Agent for Symmetrix | Configuration_00 | Symmetrix | 000184600300 | Enabled | |
| Host Agent for Windows | Discovery_00 | Host | LOSAT044 | Enabled | |
| Storage Agent for Symmetrix | Historical Data_00 | Host | LOSAT044 | Enabled | |
| Storage Agent for Symmetrix | Local Discovery_00 | Host | LOSAT044 | Enabled | |
| Storage Agent for Symmetrix | Performance Statistics_00 | Symmetrix | 000184600300 | Enabled | |
| Storage Agent for Symmetrix | Real-time BCV/RDF Status_00 | Symmetrix | 000184600300 | Enabled | |
| Storage Agent for Symmetrix | WLA Analyst_ana | Symmetrix | 000184600300 | Enabled | |
| Host Agent for Windows | WLA Analyst_ana | Host | LOSAT044 | Enabled | |
| Storage Agent for Symmetrix | WLA Daily_00 | Symmetrix | 000184600300 | Enabled | |
| Host Agent for Windows | WLA Daily_00 | Host | LOSAT044 | Enabled | |
| Storage Agent for Symmetrix | WLA Revolving_00 | Symmetrix | 000184600300 | Enabled | |
| Host Agent for Windows | WLA Revolving_00 | Host | | ed | |

**Figure 1-8 ControlCenter Policy View to Get Revolving Data**

Once selected, the revolving data for the specific data provider will be visible in the data selection window of Performance View under revolving collections.

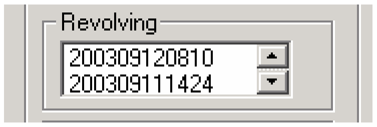


**Figure 1-9 Revolving Data in Performance View**

A revolving collection for a specific data provider is automatically named using the date and starting time that the collection was requested. For example, if the data was requested at 3 p.m., the time-stamp name of the revolving file will indicate 1 p.m. if the duration policy is 2 hours. If you request another view at 3:05, there is another file indicating a start time of 1:05.

## 1.3.3 Analyst Collection

An Analyst collection refers to a data collection session that is started by an individual for special purposes such as collecting data while a particular job is executing. Such a collection can be scheduled to begin at a certain time and continue for a specific time period. The scheduling can be specified ahead of time, for instance, an analyst can schedule a collection at night when the job runs but the analyst is not there.

Analyst data collections are explicitly named by the user as well as appended with date and time of collection:

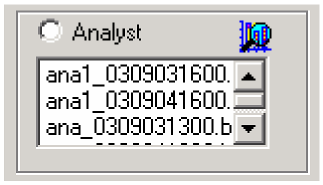


**Figure 1-10 Analyst Data Collections**

## 1.4 Private Data

In addition to the policy-driven collections noted earlier, you can store data exported to or imported from other archives. Such data resides in a directory not attached to the official archives, and is accessed via the Private Data area of the Data Selected dialog box.

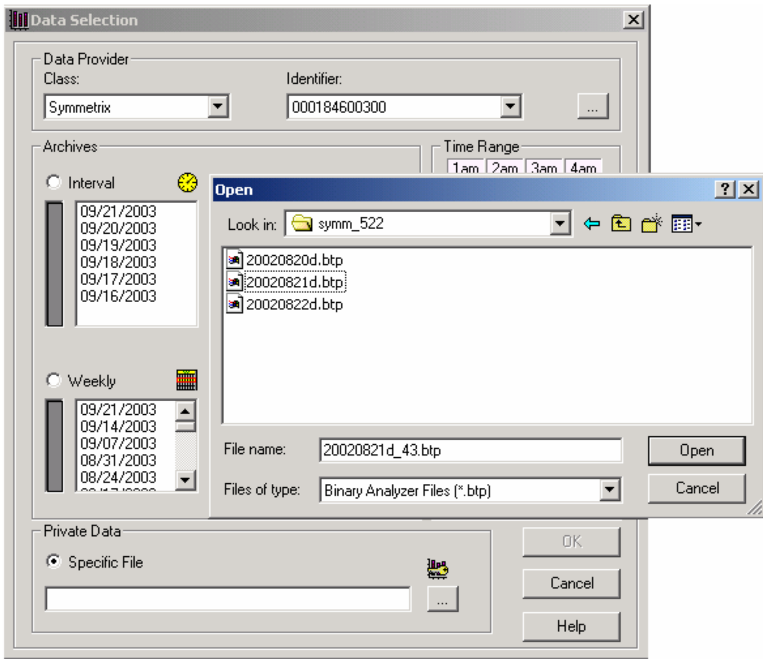


Figure 1-11 **Accessing Private Data**

This feature allows sharing of data files among Performance View users, whether its between EMC customers and EMC service personnel or any combination of such users. This allows remote analysis of data without providing full access to the local archives.

## 1.5 Automated Reports

In addition to updating the archives on a daily basis, the WLA creates a set of reports made up of predefined views. You can view these reports using the Reports menu in WLA/Performance View.

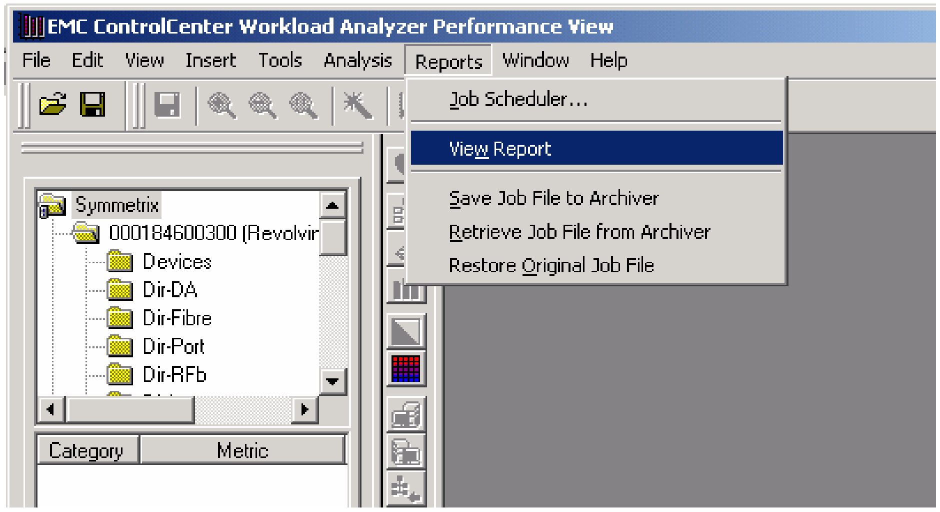


**Figure 1-12 Reports of Predefined Views**

Two types of reports are available:

- Factory Reports
- Job Scheduler

---

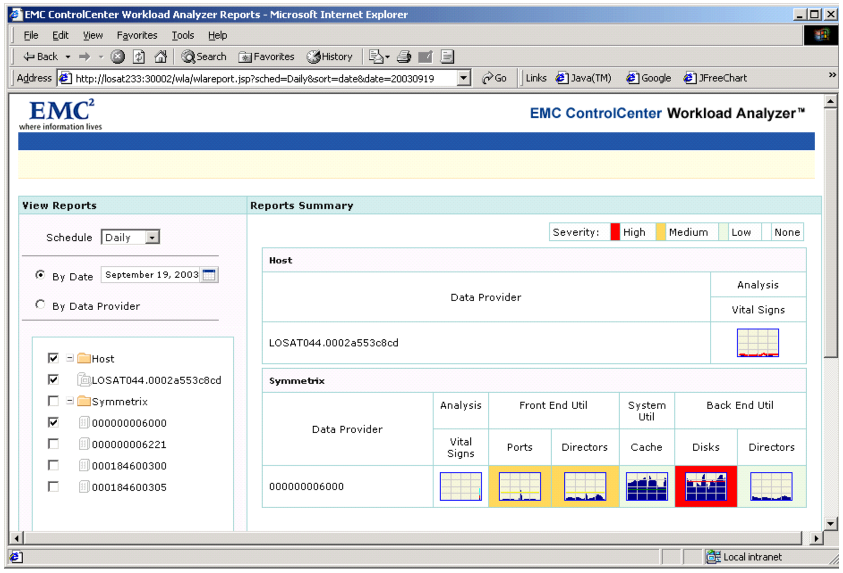
It is not the intent of this section to present the details of generating such reports. The intent here is to develop awareness of these reports. Refer to the ControlCenter user guide for additional information.

---

### 1.5.1 Factory Reports

Factory Reports are EMC-defined views that come with the WLA. These views come with pre-set threshold values for you to see and immediately recognize that a threshold was exceeded somewhere within this report category.

The report first appears as a set of thumbnail pictures with green, yellow, or red borders. These colors indicate that a threshold was crossed and is visible within the set of views of this particular thumbnail picture.


EMC ControlCenter Workload Analyzer Reports - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Links Java(TM) Google JFreeChart
EMC² where information lives
EMC ControlCenter Workload Analyzer™
Local intranet

Address http://losat233:30002/wla/wlareport.jsp?sched=Daily&sort=date&date=20030919

**View Reports**

Schedule Daily

(●) By Date September 19, 2003

( ) By Data Provider

- [✓] Host
  - [✓] LOSAT044.0002a553c8cd
- [ ] Symmetrix
  - [✓] 000000006000
  - [ ] 000000006221
  - [ ] 000184600300
  - [ ] 000184600305

**Reports Summary**

Severity: 🟥 High 🟨 Medium Low None

**Host**

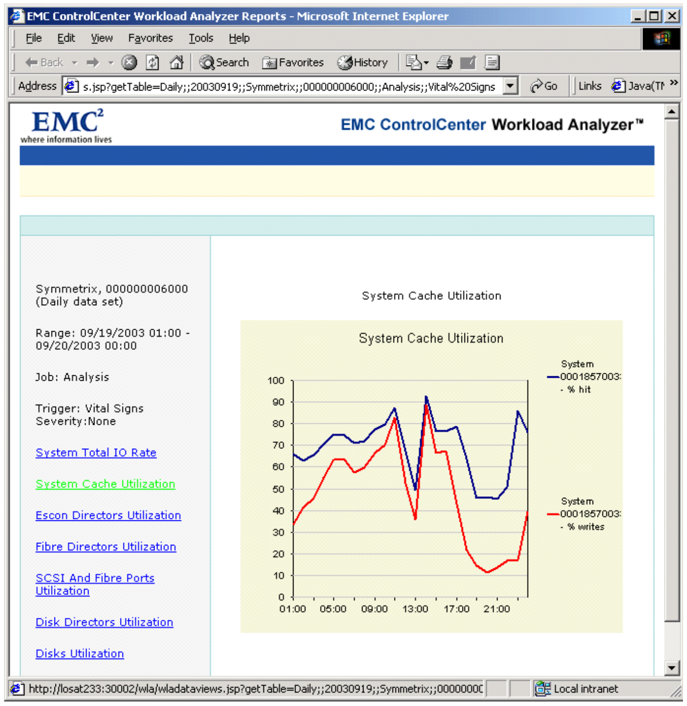
| Data Provider | Analysis |
|---|---|
| | Vital Signs |
| LOSAT044.0002a553c8cd | ▦ |

**Symmetrix**

| Data Provider | Analysis | Front End Util | | System Util | Back End Util | |
|---|---|---|---|---|---|---|
| | Vital Signs | Ports | Directors | Cache | Disks | Directors |
| 000000006000 | ▦ | 🟨 | 🟨 | ▦ | 🟥 | ▦ |

Figure 1-13 Factory Reports

Clicking the thumbnail brings up a separate window containing the predefined views that show the specific metrics that indicate a problem. The list of views is on the left and the graph picture changes as you move the cursor over the list of titles.


EMC ControlCenter Workload Analyzer Reports - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Links Java(TM)
EMC² where information lives
EMC ControlCenter Workload Analyzer™
System Cache Utilization
System Cache Utilization
100
90
80
70
60
50
40
30
20
10
0
01:00
05:00
09:00
13:00
17:00
21:00
System 0001857003: - % hit
System 0001857003: - % writes
Local intranet

Address s.jsp?getTable=Daily;;20030919;;Symmetrix;;000000006000;;Analysis;;Vital%20Signs

Symmetrix, 000000006000
(Daily data set)

Range: 09/19/2003 01:00 - 09/20/2003 00:00

Job: Analysis

Trigger: Vital Signs
Severity:None

System Total IO Rate

System Cache Utilization

Escon Directors Utilization

Fibre Directors Utilization

SCSI And Fibre Ports Utilization

Disk Directors Utilization

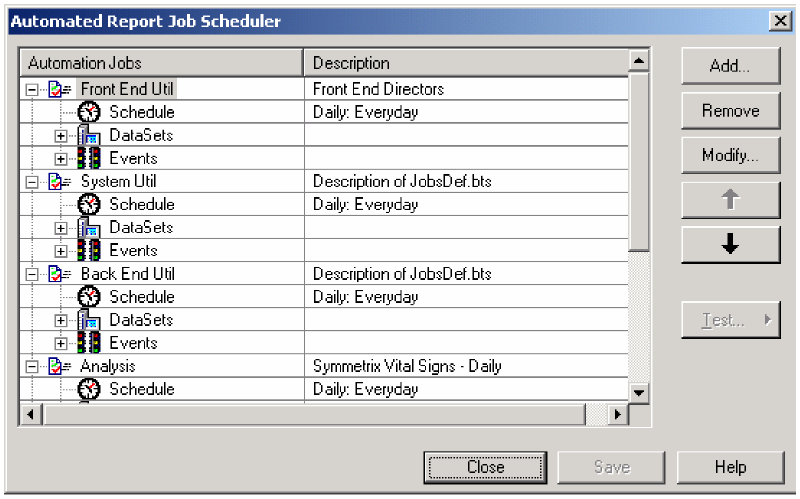
Disks Utilization

http://losat233:30002/wla/wladataviews.jsp?getTable=Daily;;20030919;;Symmetrix;;0000000

Figure 1-14 Display Predefined Views of Specific Metrics

## 1.5.2 Job Scheduler

The Job Scheduler button on the Reports menu list of Performance View brings up the following window that lets you define and control your own views in addition to the factory ones.

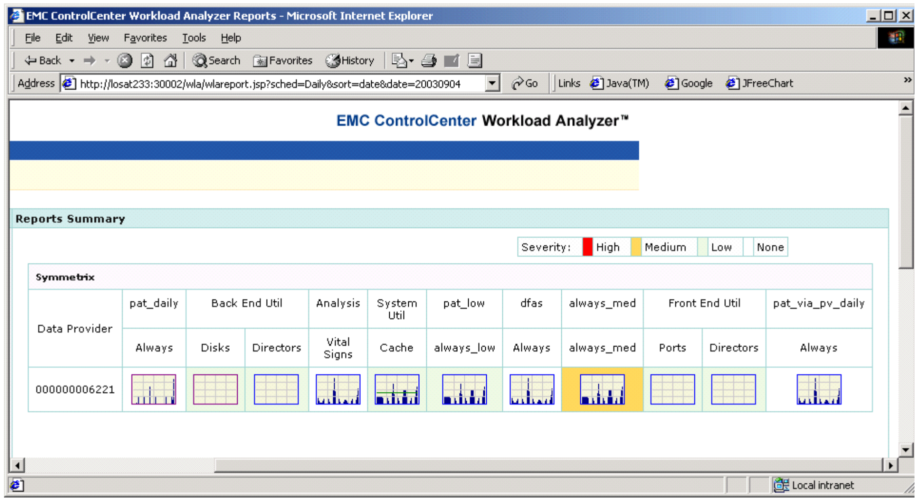


Automated Report Job Scheduler

| Automation Jobs | Description |
|---|---|
| Front End Util | Front End Directors |
| Schedule | Daily: Everyday |
| DataSets | |
| Events | |
| System Util | Description of JobsDef.bts |
| Schedule | Daily: Everyday |
| DataSets | |
| Events | |
| Back End Util | Description of JobsDef.bts |
| Schedule | Daily: Everyday |
| DataSets | |
| Events | |
| Analysis | Symmetrix Vital Signs - Daily |
| Schedule | Daily: Everyday |

Figure 1-15 **Define and Control Views**

Once user-defined views are generated, you will see them in the daily automated reports, as shown in the following example. Notice, there are more categories than the basic six factory groups previously shown.



EMC ControlCenter Workload Analyzer Reports - Microsoft Internet Explorer

Address http://losat233:30002/wla/wlareport.jsp?sched=Daily&sort=date&date=20030904

EMC ControlCenter Workload Analyzer™

Reports Summary

Severity: 🟥 High 🟨 Medium Low None

Symmetrix

| Data Provider | pat_daily | Back End Util | | Analysis | System Util | pat_low | dfas | always_med | Front End Util | | pat_via_pv_daily |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Always | Disks | Directors | Vital Signs | Cache | always_low | Always | always_med | Ports | Directors | Always |
| 000000006221 | | | | | | | | | | | |

Figure 1-16 **Daily Automated Reports**

## 1.6 Performance View

Performance View (PV) is the manual ad hoc analysis tool used to view the performance data stored in the Workload Analyzer archives. It is a feature-rich tool that allows you to create a multitude of views of the data in order to more easily identify performance problems.

There are several key features in the product that you should familiarize yourself with in order to become more proficient in the data mining techniques when diagnosing problems. This section highlights the three modes of data viewing available via PV and briefly identify a few not-so-obvious features of the tool.

To start, open a data file. The left-hand pane shows a pane with three tabs:

- Views
- Metrics
- Config

The Views tab is the one that opens up automatically at the start. Familiarizing yourself with the contents of each of these tabs and their interrelationship is the start of becoming a seasoned user of the tool.

### 1.6.1 Views Tab

Views are programmed graphs with definitions and characteristics that are stored in a file and are loaded when the data file opens. When the product is first installed, the visible views are the ones that were programmed at EMC engineering prior to shipping of the product.

When views are created (whether EMC- or user-defined), they may be assigned a characteristic known as “vital sign” and shown in the Views tab with the red heart icon. This allows important metrics to be accessed quickly.

Vital Sign graphs can be displayed individually by double-clicking the graph label, or as a group by highlighting the data provider ID, or any component category belonging to that data provider as long as there are views defined in that category with the Vital Sign icon, and then using the red-heart tool on the vertical tool bar of Performance View.

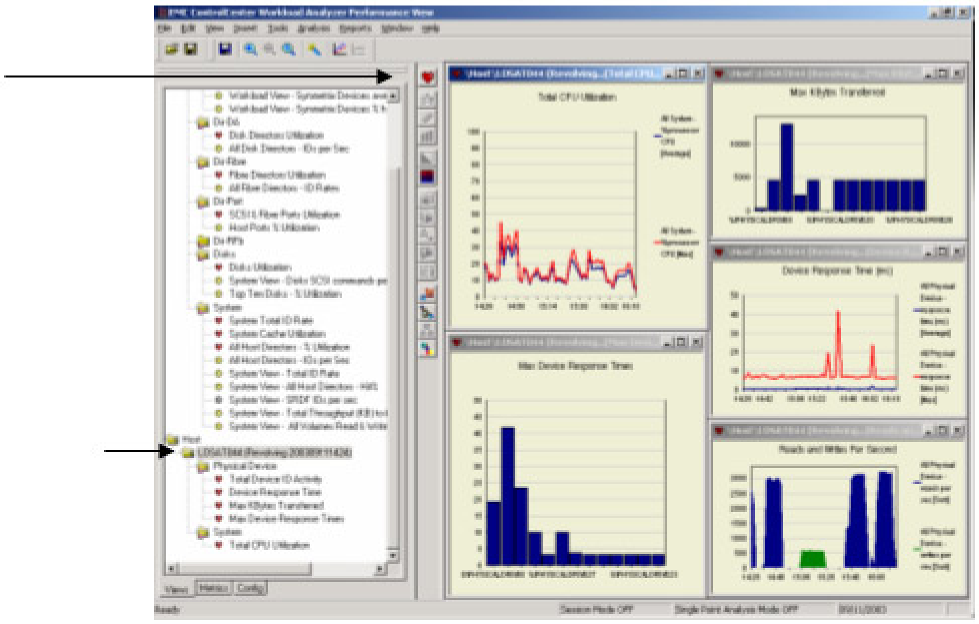
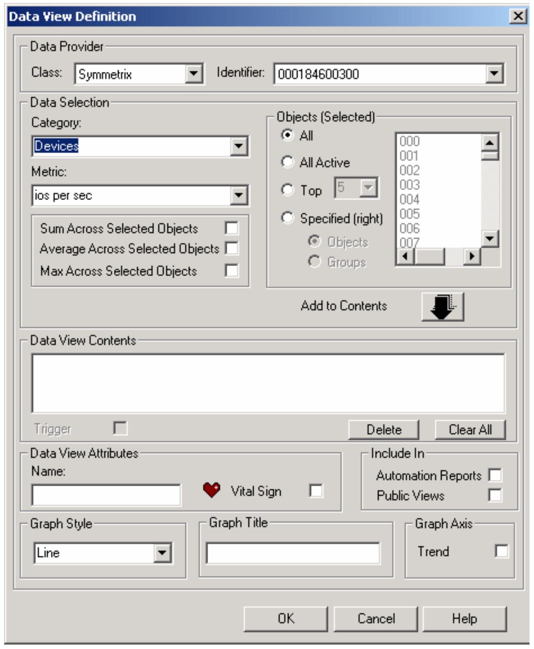

Total CPU Utilization
Max KBytes Transferred
Device Response Time (ms)
Max Device Response Times
Reads and Writes Per Second

Figure 1-17 Vital Sign Graphs

You can create and define views and store them in the same way as the factory-provided views that come with the product. Creating such views simplifies daily analysis, and these views can be included in the user-defined automated reports.

You can right-click or select the menu item Insert/New Data View to bring up the following screen.


Data View Definition

Data Provider
Class: Symmetrix
Identifier: 000184600300

Data Selection
Category: Devices
Metric: ios per sec
Sum Across Selected Objects [ ]
Average Across Selected Objects [ ]
Max Across Selected Objects [ ]

Objects (Selected)
[✓ All] All Active Top 5 Specified (right)
Objects Groups
000 001 002 003 004 005 006 007

Add to Contents

Data View Contents
Trigger [ ]

Data View Attributes
Name:
Vital Sign [ ]

Include In
Automation Reports [ ]
Public Views [ ]

Graph Style: Line
Graph Title:
Graph Axis: Trend [ ]

Figure 1-18 Create, Define, and Store Views

The highlighted capabilities of this screen are:

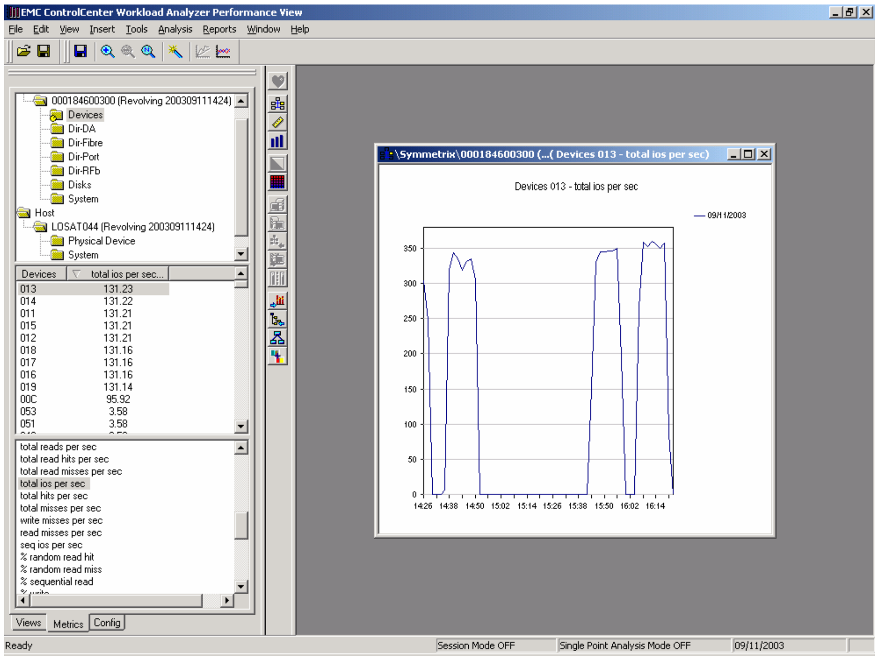
- Generate graphs for specific data providers or for all data providers of a specific type.
- Define subsets of component categories such as all objects, top or predefined groups
- Specify lines to represent the sum, average, or maximum value of the selected metric.
- Select graph styles (line, bar, area, etc.)
- Combine categories
- Identify view as vital signs.
- Include in automated reports

You are encouraged to experiment with views creation since it simplifies the monitoring and diagnosis process for daily applications and workloads. In addition to the previous view wizard, you can also create views from graphs created via the Metrics tab.

One limitation of the Views feature is that it can be created and saved for a single data provider only. That is, a graph that combines data from multiple data providers (two or more Symmetrix arrays, a Symmetrix array and a host, etc.) cannot be re-created automatically. Such graphs are limited to ad hoc analysis only.

## 1.6.2 Metrics Tab

Most of the ad-hoc analysis of data is done using Metrics tab. Functions performed here are creating the graph, modifying the graph, saving the graph as a view, navigating to/from the configuration and links views to see relationships among components, and generally conducting detailed investigations.


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
\Symmetrix\000184600300 (...( Devices 013 - total ios per sec)
Devices 013 - total ios per sec
09/11/2003
350
300
250
200
150
100
50
0
14:26 14:38 14:50 15:02 15:14 15:26 15:38 15:50 16:02 16:14
Views Metrics Config
Ready
Session Mode OFF
Single Point Analysis Mode OFF
09/11/2003

000184600300 (Revolving 200309111424)
- Devices
- Dir-DA
- Dir-Fibre
- Dir-Port
- Dir-RFb
- Disks
- System

Host
- LOSAT044 (Revolving 200309111424)
  - Physical Device
  - System

| Devices | total ios per sec... |
|---|---|
| 013 | 131.23 |
| 014 | 131.22 |
| 011 | 131.21 |
| 015 | 131.21 |
| 012 | 131.21 |
| 018 | 131.16 |
| 017 | 131.16 |
| 016 | 131.16 |
| 019 | 131.14 |
| 00C | 95.92 |
| 053 | 3.58 |
| 051 | 3.58 |

total reads per sec
total read hits per sec
total read misses per sec
total ios per sec
total hits per sec
total misses per sec
write misses per sec
read misses per sec
seq ios per sec
% random read hit
% random read miss
% sequential read

Figure 1-19 **Ad-hoc Data Analysis from Metrics Tab**

### 1.6.2.1 Graph Creation

You create a graph by selecting a category, choosing one or more instances of that category, and double-clicking a metric. You can modify the graph in multiple ways. It can include multiple lines that can be added to it using a tool icon (see users guide for details). This feature allows you to put the data from multiple categories or data providers on a single graph.

### 1.6.2.2 Graph Modification

Any graph can be modified using the Graph Wizard tool:

Figure 1-20 shows the graph Wizard screen.

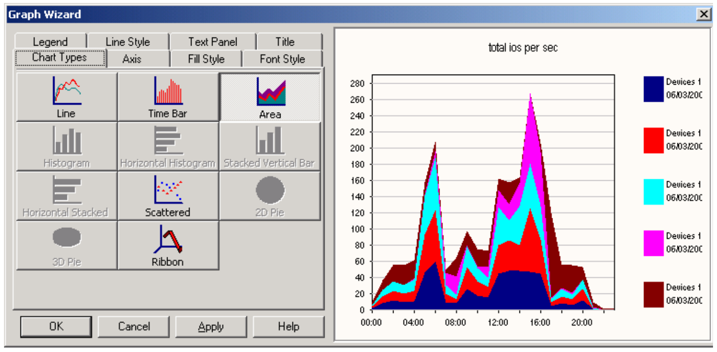


**Figure 1-20 Graph Wizard Screen**

Figure 1-21 shows four different views of the same data after each graph was modified using the graph wizard.

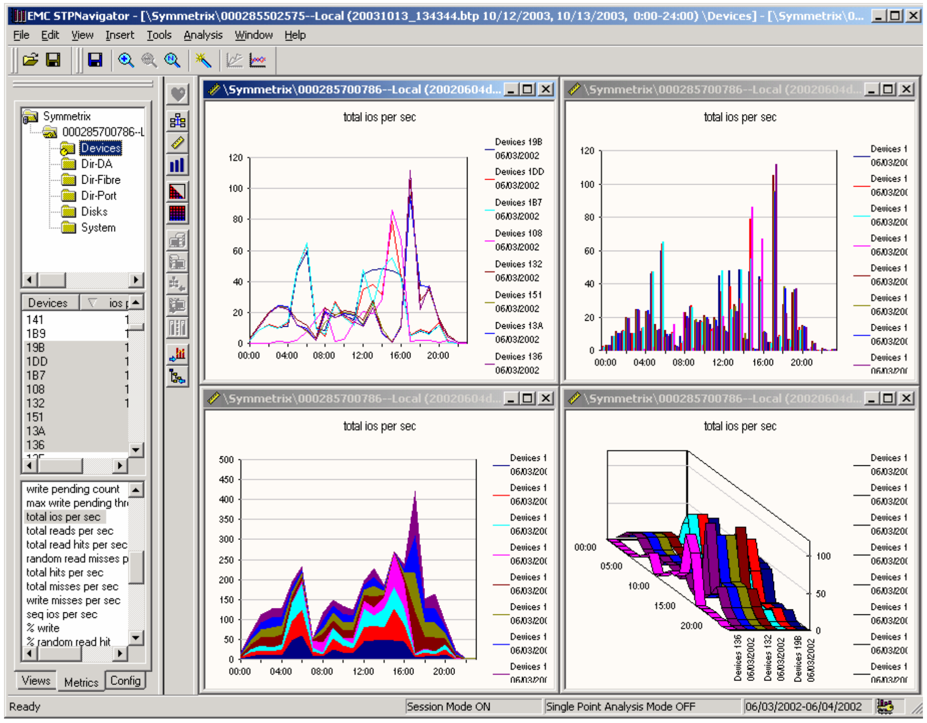
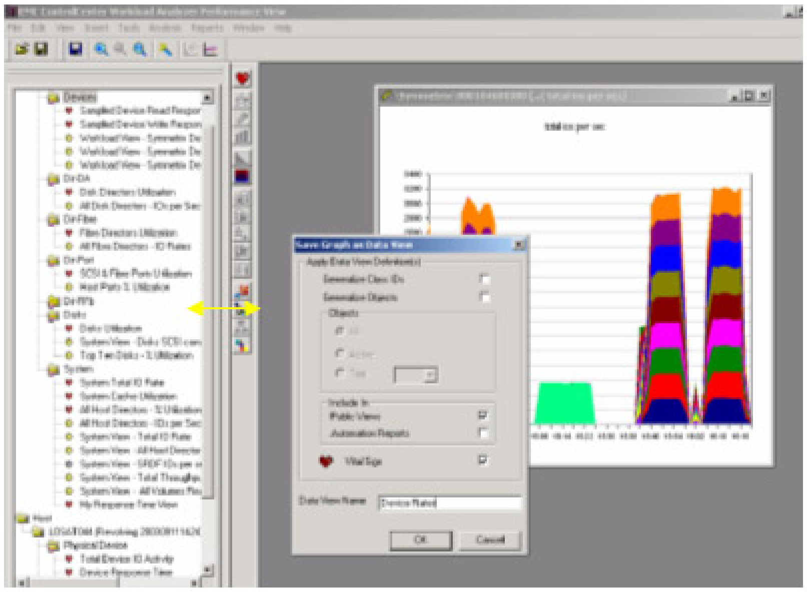


**Figure 1-21 Four Views of the Same Data**

A graph can be interpreted in different ways. Depending on the desired message, a different rendering of the graph will provide different information. The graph wizard lets you focus on the message presented by the data.

### 1.6.2.3 Saving Graph as a View

When performing analysis for a single data provider, you can save selected graphs as views without having to re-create them in the Views tab. Any graph on display that contains data from a single data provider can be stored the next time any data from that data provider is opened, and can also be used in the automated reports for that data provider. As mentioned before, the only limitation on this feature is that data from multiple providers cannot be combined and saved as a view.



Apply Data View Definition(s)

Generation Class IDs ☐

Generation Objects ☐

Objects: (● All) ○ Active ○ Top

Include In

Public Views ☑

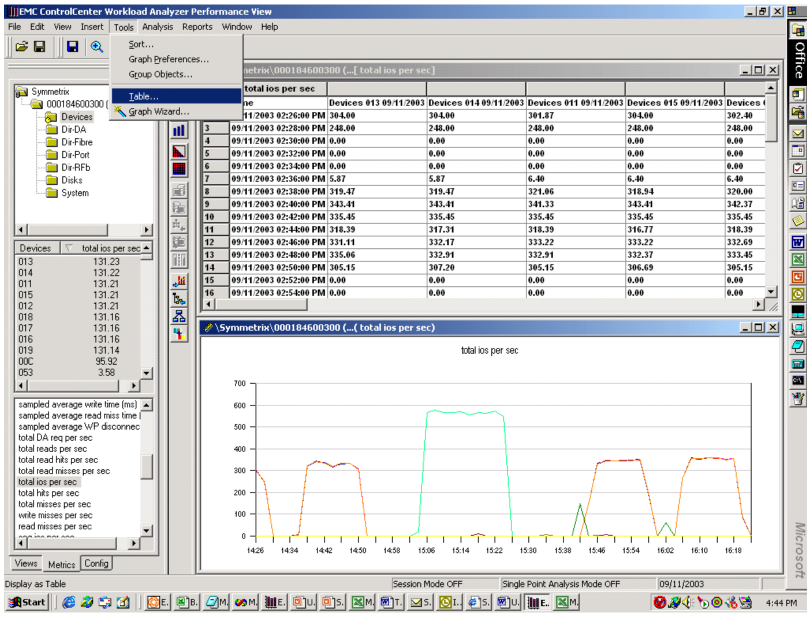
Automation Reports ☐

VitalSign ☑

Data View Name: Device Rates

Figure 1-22

Data can be displayed in a tabular format. Select **Tools, Table** to display the table for the selected graph.



| Devices | total ios per sec |
|---|---|
| 013 | 131.23 |
| 014 | 131.22 |
| 011 | 131.21 |
| 015 | 131.21 |
| 012 | 131.21 |
| 018 | 131.16 |
| 017 | 131.16 |
| 016 | 131.16 |
| 019 | 131.14 |
| 00C | 95.92 |
| 053 | 3.58 |

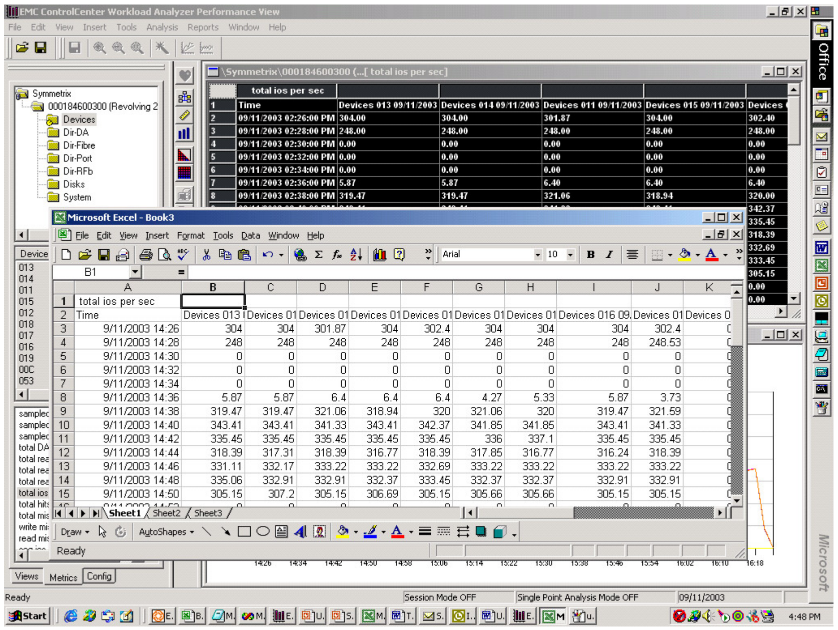
sampled average write time (ms)
sampled average read miss time
sampled average WP disconnec
total DA req per sec
total reads per sec
total read hits per sec
total read misses per sec
total ios per sec
total hits per sec
total misses per sec
write misses per sec
read misses per sec

metrix\000184600300 (...[ total ios per sec]

| | total ios per sec | | | | | |
|---|---|---|---|---|---|---|
| | ne | Devices 013 09/11/2003 | Devices 014 09/11/2003 | Devices 011 09/11/2003 | Devices 015 09/11/2003 | Devices |
| | 11/2003 02:26:00 PM | 304.00 | 304.00 | 301.87 | 304.00 | 302.40 |
| 3 | 09/11/2003 02:28:00 PM | 248.00 | 248.00 | 248.00 | 248.00 | 248.00 |
| 4 | 09/11/2003 02:30:00 PM | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 5 | 09/11/2003 02:32:00 PM | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 6 | 09/11/2003 02:34:00 PM | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 7 | 09/11/2003 02:36:00 PM | 5.87 | 5.87 | 6.40 | 6.40 | 6.40 |
| 8 | 09/11/2003 02:38:00 PM | 319.47 | 319.47 | 321.06 | 318.94 | 320.00 |
| 9 | 09/11/2003 02:40:00 PM | 343.41 | 343.41 | 341.33 | 343.41 | 342.37 |
| 10 | 09/11/2003 02:42:00 PM | 335.45 | 335.45 | 335.45 | 335.45 | 335.45 |
| 11 | 09/11/2003 02:44:00 PM | 318.39 | 317.31 | 318.39 | 316.77 | 318.39 |
| 12 | 09/11/2003 02:46:00 PM | 331.11 | 332.17 | 333.22 | 333.22 | 332.69 |
| 13 | 09/11/2003 02:48:00 PM | 335.06 | 332.91 | 332.91 | 332.37 | 333.45 |
| 14 | 09/11/2003 02:50:00 PM | 305.15 | 307.20 | 305.15 | 306.69 | 305.15 |
| 15 | 09/11/2003 02:52:00 PM | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 16 | 09/11/2003 02:54:00 PM | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |

Figure 1-23 **Display Data in Tabular Format**

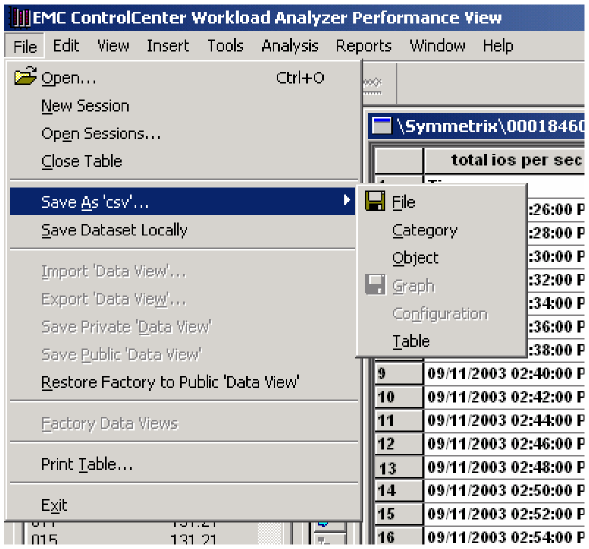
If you want to export the data that appears on a graph, the easiest way to do so is through a cut/paste mechanism. By clicking the upper-left-hand cell in the table, the complete table is selected. CTRL-C will copy the table to the clip board, while a CTRL-Z will paste the table into an Excel spreadsheet. In addition, the user has the option to save the data in a file. Furthermore, the user can designate the scope of the data to be saved. The scope is designated from the whole data set to specific categories or objects as well as the data associated with a specific graph or table. Lastly, the configuration of the data provider (for Symmetrix and Open System hosts only) can also be saved as a .csv file. Figure 1-24 shows an example of a table view within Performance View along with the pasted view of an Excel spreadsheet.


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000184600300 (Revolving 2
Devices
Dir-DA
Dir-Fibre
Dir-Port
Dir-RFb
Disks
System
Views Metrics Config
\Symmetrix\000184600300 (...[ total ios per sec]
Microsoft Excel - Book3
File Edit View Insert Format Tools Data Window Help
Arial 10
B1
Sheet1 Sheet2 Sheet3
Draw AutoShapes
Ready
Session Mode OFF
Single Point Analysis Mode OFF
09/11/2003
Start
4:48 PM
Office
Microsoft

| | total ios per sec | | | | | |
|---|---|---|---|---|---|---|
| 1 | Time | Devices 013 09/11/2003 | Devices 014 09/11/2003 | Devices 011 09/11/2003 | Devices 015 09/11/2003 | Devices |
| 2 | 09/11/2003 02:26:00 PM | 304.00 | 304.00 | 301.87 | 304.00 | 302.40 |
| 3 | 09/11/2003 02:28:00 PM | 248.00 | 248.00 | 248.00 | 248.00 | 248.00 |
| 4 | 09/11/2003 02:30:00 PM | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 5 | 09/11/2003 02:32:00 PM | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 6 | 09/11/2003 02:34:00 PM | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 7 | 09/11/2003 02:36:00 PM | 5.87 | 5.87 | 6.40 | 6.40 | 6.40 |
| 8 | 09/11/2003 02:38:00 PM | 319.47 | 319.47 | 321.06 | 318.94 | 320.00 |

| | A | B | C | D | E | F | G | H | I | J | K |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | total ios per sec | | | | | | | | | | |
| 2 | Time | Devices 013 | Devices 01 | Devices 01 | Devices 01 | Devices 01 | Devices 01 | Devices 01 | Devices 016 09 | Devices 01 | Devices 0 |
| 3 | 9/11/2003 14:26 | 304 | 304 | 301.87 | 304 | 302.4 | 304 | 304 | 304 | 302.4 | 0 |
| 4 | 9/11/2003 14:28 | 248 | 248 | 248 | 248 | 248 | 248 | 248 | 248 | 248.53 | 0 |
| 5 | 9/11/2003 14:30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 9/11/2003 14:32 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 9/11/2003 14:34 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 9/11/2003 14:36 | 5.87 | 5.87 | 6.4 | 6.4 | 6.4 | 4.27 | 5.33 | 5.87 | 3.73 | 0 |
| 9 | 9/11/2003 14:38 | 319.47 | 319.47 | 321.06 | 318.94 | 320 | 321.06 | 320 | 319.47 | 321.59 | 0 |
| 10 | 9/11/2003 14:40 | 343.41 | 343.41 | 341.33 | 343.41 | 342.37 | 341.85 | 341.85 | 343.41 | 341.33 | 0 |
| 11 | 9/11/2003 14:42 | 335.45 | 335.45 | 335.45 | 335.45 | 335.45 | 336 | 337.1 | 335.45 | 335.45 | 0 |
| 12 | 9/11/2003 14:44 | 318.39 | 317.31 | 318.39 | 316.77 | 318.39 | 317.85 | 316.77 | 316.24 | 318.39 | 0 |
| 13 | 9/11/2003 14:46 | 331.11 | 332.17 | 333.22 | 333.22 | 332.69 | 333.22 | 333.22 | 333.22 | 333.22 | 0 |
| 14 | 9/11/2003 14:48 | 335.06 | 332.91 | 332.91 | 332.37 | 333.45 | 332.37 | 332.37 | 332.91 | 332.91 | 0 |
| 15 | 9/11/2003 14:50 | 305.15 | 307.2 | 305.15 | 306.69 | 305.15 | 305.66 | 305.66 | 305.15 | 305.15 | 0 |

Figure 1-24



| | |
|---|---|
| 9 | 09/11/2003 02:40:00 P |
| 10 | 09/11/2003 02:42:00 P |
| 11 | 09/11/2003 02:44:00 P |
| 12 | 09/11/2003 02:46:00 P |
| 13 | 09/11/2003 02:48:00 P |
| 14 | 09/11/2003 02:50:00 P |
| 15 | 09/11/2003 02:52:00 P |
| 16 | 09/11/2003 02:54:00 P |

Figure 1-25

## 1.6.3 Working with the Configuration Views

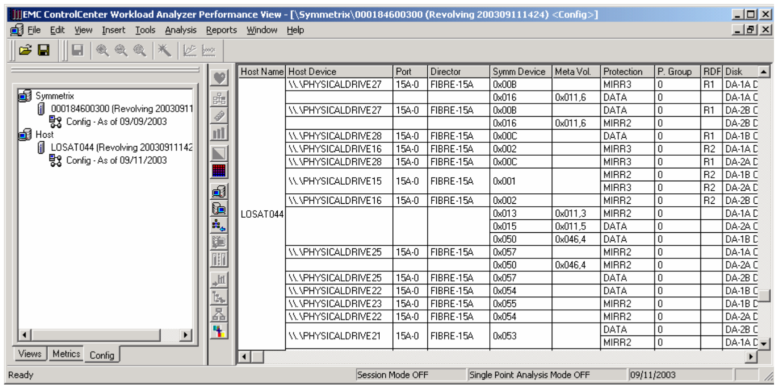
The configuration of the storage array describes the relationship among logical and physical components in the system. Within the Symmetrix, devices are the logical components and directors' busses and disks are the physical components. The relationships among these components define the I/O paths taken to serve an I/O command.

When you view performance metrics, there are questions such as:

- Which devices make up a meta group?
- Who are the members of each RAID group?

- What devices reside on the disk?
- What are the host-to-storage device mappings?
- Which ports are connected to which hosts?
- How can you quickly isolate and view the performance metrics of any of the above sets?

Using Performance View, there are two ways to view configuration data: the links view and the tabular view. The Config tab of Performance View is the method used to display the tabular view of the configuration of the open data sets. This view applies only to Symmetrix and open systems hosts.

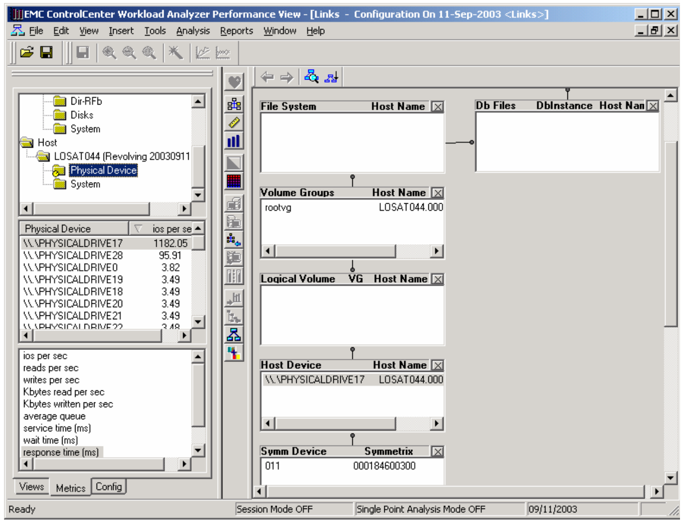

EMC ControlCenter Workload Analyzer Performance View - [\Symmetrix\000184600300 (Revolving 200309111424) <Config>]
File
Edit
View
Insert
Tools
Analysis
Reports
Window
Help
Views
Metrics
Config

Symmetrix
000184600300 (Revolving 20030911
Config - As of 09/09/2003
Host
LOSAT044 (Revolving 20030911142
Config - As of 09/11/2003

| Host Name | Host Device | Port | Director | Symm Device | Meta Vol. | Protection | P. Group | RDF | Disk |
|---|---|---|---|---|---|---|---|---|---|
| LOSAT044 | \\.\PHYSICALDRIVE27 | 15A-0 | FIBRE-15A | 0x00B | | MIRR3 | 0 | R1 | DA-1A C |
| | | | | 0x016 | 0x011,6 | DATA | 0 | | DA-1A C |
| | \\.\PHYSICALDRIVE27 | 15A-0 | FIBRE-15A | 0x00B | | DATA | 0 | R1 | DA-2B C |
| | | | | 0x016 | 0x011,6 | MIRR2 | 0 | | DA-2B C |
| | \\.\PHYSICALDRIVE28 | 15A-0 | FIBRE-15A | 0x00C | | DATA | 0 | R1 | DA-1B C |
| | \\.\PHYSICALDRIVE16 | 15A-0 | FIBRE-15A | 0x002 | | MIRR3 | 0 | R2 | DA-1A C |
| | \\.\PHYSICALDRIVE28 | 15A-0 | FIBRE-15A | 0x00C | | MIRR3 | 0 | R1 | DA-2A C |
| | \\.\PHYSICALDRIVE15 | 15A-0 | FIBRE-15A | 0x001 | | MIRR2 | 0 | R2 | DA-1B C |
| | | | | | | MIRR3 | 0 | R2 | DA-2A C |
| | \\.\PHYSICALDRIVE16 | 15A-0 | FIBRE-15A | 0x002 | | MIRR2 | 0 | R2 | DA-2B C |
| | | | | 0x013 | 0x011,3 | MIRR2 | 0 | | DA-1A C |
| | | | | 0x015 | 0x011,5 | DATA | 0 | | DA-2A C |
| | | | | 0x050 | 0x046,4 | DATA | 0 | | DA-1B C |
| | \\.\PHYSICALDRIVE25 | 15A-0 | FIBRE-15A | 0x057 | | MIRR2 | 0 | | DA-1A C |
| | | | | 0x050 | 0x046,4 | MIRR2 | 0 | | DA-2A C |
| | \\.\PHYSICALDRIVE25 | 15A-0 | FIBRE-15A | 0x057 | | DATA | 0 | | DA-2B C |
| | \\.\PHYSICALDRIVE22 | 15A-0 | FIBRE-15A | 0x054 | | DATA | 0 | | DA-1B C |
| | \\.\PHYSICALDRIVE23 | 15A-0 | FIBRE-15A | 0x055 | | MIRR2 | 0 | | DA-1B C |
| | \\.\PHYSICALDRIVE22 | 15A-0 | FIBRE-15A | 0x054 | | MIRR2 | 0 | | DA-2A C |
| | \\.\PHYSICALDRIVE21 | 15A-0 | FIBRE-15A | 0x053 | | DATA | 0 | | DA-2B C |
| | | | | | | MIRR2 | 0 | | DA-1A C |

Ready | Session Mode OFF | Single Point Analysis Mode OFF | 09/11/2003

**Figure 1-26 Tabular View of Configuration**


EMC ControlCenter Workload Analyzer Performance View - [Links - Configuration On 11-Sep-2003 <Links>]
File
Edit
View
Insert
Tools
Analysis
Reports
Window
Help
Views
Metrics
Config

Dir-RFb
Disks
System
Host
LOSAT044 (Revolving 20030911
Physical Device
System

| Physical Device | ios per se |
|---|---|
| \\.\PHYSICALDRIVE17 | 1182.05 |
| \\.\PHYSICALDRIVE28 | 95.91 |
| \\.\PHYSICALDRIVE0 | 3.82 |
| \\.\PHYSICALDRIVE19 | 3.49 |
| \\.\PHYSICALDRIVE18 | 3.49 |
| \\.\PHYSICALDRIVE20 | 3.49 |
| \\.\PHYSICALDRIVE21 | 3.49 |
| \\.\PHYSICALDRIVE22 | 3.48 |

ios per sec
reads per sec
writes per sec
Kbytes read per sec
Kbytes written per sec
average queue
service time (ms)
wait time (ms)
response time (ms)

| File System | Host Name |
|---|---|

| Db Files | DbInstance | Host Nan |
|---|---|---|

| Volume Groups | Host Name |
|---|---|
| rootvg | LOSAT044.000 |

| Logical Volume | VG | Host Name |
|---|---|---|

| Host Device | Host Name |
|---|---|
| \\.\PHYSICALDRIVE17 | LOSAT044.000 |

| Symm Device | Symmetrix |
|---|---|
| 011 | 000184600300 |

Ready | Session Mode OFF | Single Point Analysis Mode OFF | 09/11/2003

**Figure 1-27 Links View of Configuration**

The following toolbar tools are used with configuration views:

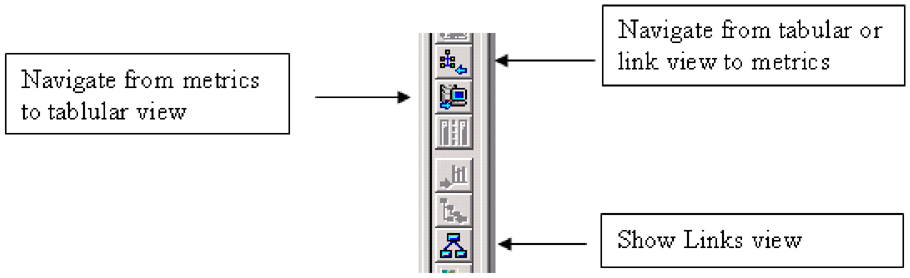

Navigate from metrics to tablular view
Navigate from tabular or link view to metrics
Show Links view

Figure 1-28 **Configuration Views Toolbar**

While there are some functional differences between these two views of the configurations, there is one important distinction: the links view is only visible when Performance View is connected to a ControlCenter Repository. When not connected, the only way to view the configuration is via the tabular view.

To see a configuration with data when not connected to a repository, you must properly extract the desired data. This means that simply copying a .btp file from the archives is not sufficient to view the configurations. To see the configuration, you must create separate isolated archives. To do so, select **File/Save Dataset Locally** while connected to the Repository and the original archives.

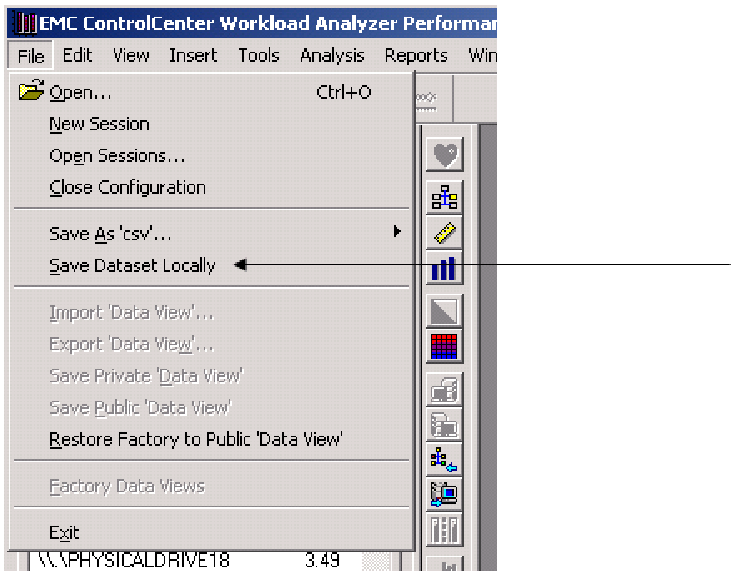


Figure 1-29

A dataset saved in this manner will contain all the necessary information to allow you to view both data and configuration information. Only the tabular configuration table will be usable in this mode.

Before viewing the data with a stand-alone version of Performance View, you must first register the saved archives with the program. Following the stars in Figure 1-30 on page 1-22 shows how you can register a properly saved data set as a local archives and view with configuration information.

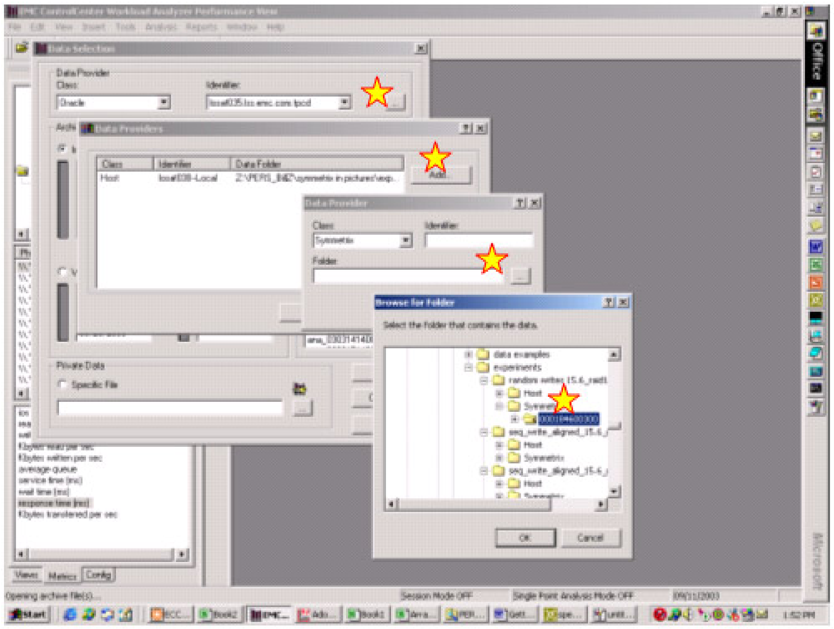


Figure 1-30

2

# The Symmetrix Architecture: A Brief Overview

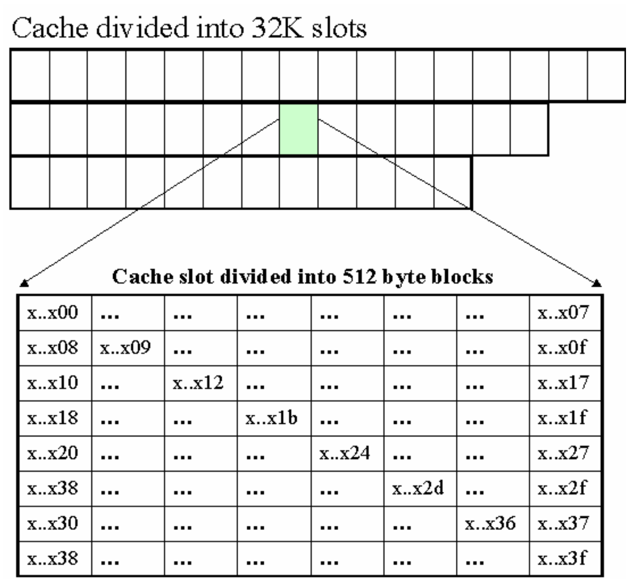
This chapter reviews the following topic:

## 2.1 The Symmetrix Cache

The Symmetrix cache, as in other storage systems, is made up of a series of memory boards, each with its own connection to the CPUs that access the RAM. In the case of the Symmetrix array, the front end and back end consist of two different types of CPUs that manage the flow of data to and from hosts and disks. A combination of hardware features, software architecture, and algorithms determine the speed and effectiveness of the system as a whole.

### 2.1.1 Cache Slots

The Symmetrix system manages cache and storage differently for mainframe operations than for open systems operations. The main difference is in the size of a cache slot used for mainframe data compared with that used for open systems data. The examples in this book use the open systems model to explain the analysis process.


Cache divided into 32K slots
Cache slot divided into 512 byte blocks

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| x..x00 | ... | ... | ... | ... | ... | ... | x..x07 |
| x..x08 | x..x09 | ... | ... | ... | ... | ... | x..x0f |
| x..x10 | ... | x..x12 | ... | ... | ... | ... | x..x17 |
| x..x18 | ... | ... | x..x1b | ... | ... | ... | x..x1f |
| x..x20 | ... | ... | ... | x..x24 | ... | ... | x..x27 |
| x..x38 | ... | ... | ... | ... | x..x2d | ... | x..x2f |
| x..x30 | ... | ... | ... | ... | ... | x..x36 | x..x37 |
| x..x38 | ... | ... | ... | ... | ... | ... | x..x3f |

**Figure 2-1 Open Systems Cache Slot = 32 KB and Divided into 64 512-Byte Blocks**

The smallest size of an I/O is always 512 bytes, which is represented by a single slice. A 4 KB I/O is represented by a set of 8 slices, and a 32 KB I/O is represented by the full 64-slice band.

An I/O command is offered to the storage system in the form of a SCSI command description block (CDB). A trace of the Symmetrix I/O operations shows a division classification of a few CDBs. Of primary interest is the address in the CDB. The address block indicates the location within a cache slot where the data begins.

| Time Stamp | Operation | Port | LUN | Address | Size (# of blocks) |
|---|---|---|---|---|---|
| 43.60926 | Write | 1b | 00CB | 36457418 | 64 |
| 43.61218 | Write | 2a | 00CB | 36465930 | 64 |
| 43.61597 | Write | 1b | 00CB | 36457482 | 64 |
| 43.61893 | Write | 2a | 00CB | 36465994 | 64 |
| 43.62284 | Write | 1b | 00CB | 36457546 | 64 |

To determine where within a cache slot the address will start, the address needs to be converted first to hexadecimal format, and then the value MOD(address,x3f) needs to be calculated.

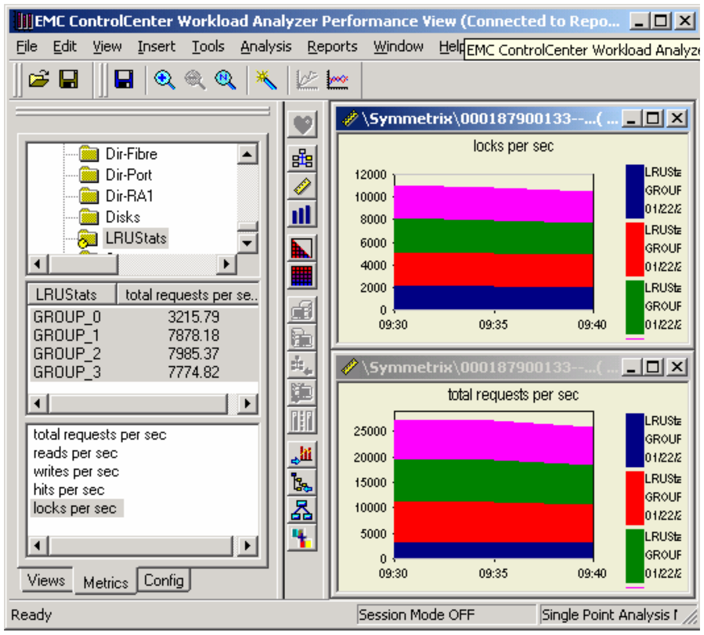
| Time Stamp | Operation | Port | LUN | Address | Cache Slot start location | Size (# of blocks) |
|---|---|---|---|---|---|---|
| 43.60926 | Write | 1b | 00CB | 36457418 | x0b | 64 |
| 43.61218 | Write | 2a | 00CB | 36465930 | x12 | 64 |
| 43.61597 | Write | 1b | 00CB | 36457482 | x0c | 64 |
| 43.61893 | Write | 2a | 00CB | 36465994 | x13 | 64 |
| 43.62284 | Write | 1b | 00CB | 36457546 | x0d | 64 |

The importance of where in the cache slot the I/O data begins will become clear in the sections on alignment and CRC.

## 2.1.2 LRU Tables

The Symmetrix system uses a Least Recently Used (LRU) mechanism to manage the allocation of cache slots to directors. The LRU tables keep track of the status of each of the available slots and the protocol used by any director to access slots is known as a *Request*. A request may contain a reference to one or more cache slots depending on the type of activity performed. We will show multiple examples of activities, and how WLA represents these activities, depending on the workload illustrated.

Generally speaking, when a director needs to write to cache, there may be a need to first lock an LRU table in order to allocate the needed cache slots. WLA collects Symmetrix LRU table statistics and presents them to the users (Figure 2-2).


EMC ControlCenter Workload Analyzer Performance View (Connected to Repo...
File Edit View Insert Tools Analysis Reports Window Help
EMC ControlCenter Workload Analyze
Dir-Fibre
Dir-Port
Dir-RA1
Disks
LRUStats
\Symmetrix\000187900133--...( ...
locks per sec
\Symmetrix\000187900133--...( ...
total requests per sec
LRUSta GROUP 01/22/
Views
Metrics
Config
Ready
Session Mode OFF
Single Point Analysis

| LRUStats | total requests per se.. |
|---|---|
| GROUP_0 | 3215.79 |
| GROUP_1 | 7878.18 |
| GROUP_2 | 7985.37 |
| GROUP_3 | 7774.82 |

total requests per sec
reads per sec
writes per sec
hits per sec
locks per sec

Figure 2-2 **Symmetrix LRU Table Statistics**

## 2.1.3 Key Cache Management Metrics

Metrics associated with cache management are presented on the Symmetrix by front-end and back-end directors and by devices. The labels used to name these metrics are somewhat confusing due to the evolution of the product.

### 2.1.3.1 Directors Cache Statistics

Symmetrix Directors report on two types of activities (among other metrics):

- **I/Os** — These count data movement commands issued from external hosts in the case of from-end directors, or issued to disk controllers in the case of back-end directors.
- **Requests** — These represent the number of accesses to cache slots that resulted in servicing an I/O operation.

Dividing activities into such categories as read-hit, write, read-miss, etc., all refer to requests when reported by directors, both front end and back end.

### 2.1.3.2 Logical Devices Cache Statistics

Symmetrix devices report on only one type of front-end activity (among other metrics)

- **I/Os** — These count data movement commands issued from external hosts in the case of from-end directors.

Dividing activities into such categories as read-hit, write, read-miss, etc., all refer to I/Os when reported by Symmetrix devices.

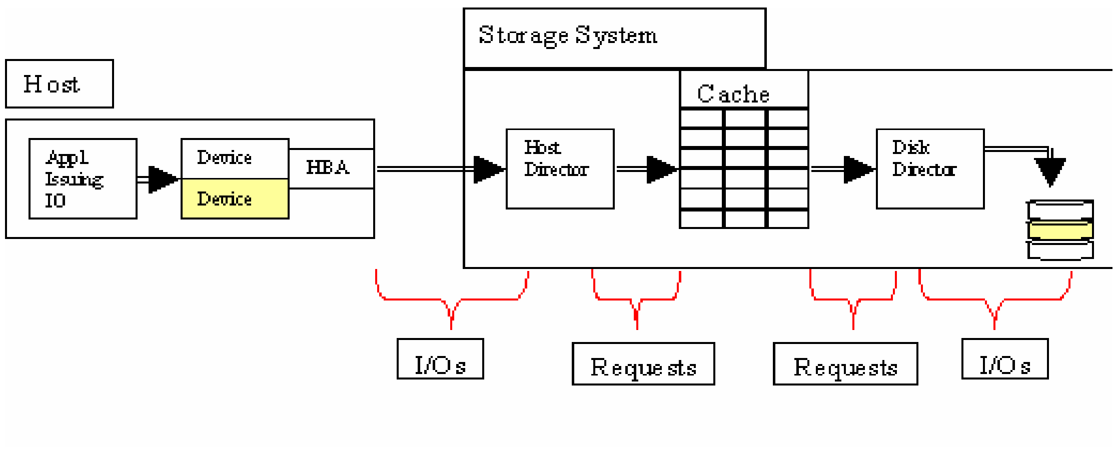


Figure 2-3

As stated earlier, most non-Symmetrix cached storage arrays do not report statistics at this level. The effect of cache management on I/O response time is significant. By understanding the methods used to manage cache you can organize data at the application level to optimize throughput and response times. The following sections cover the effects that various aspects of cache management have on I/O response times.

## 2.1.4 Cyclic Redundancy Check

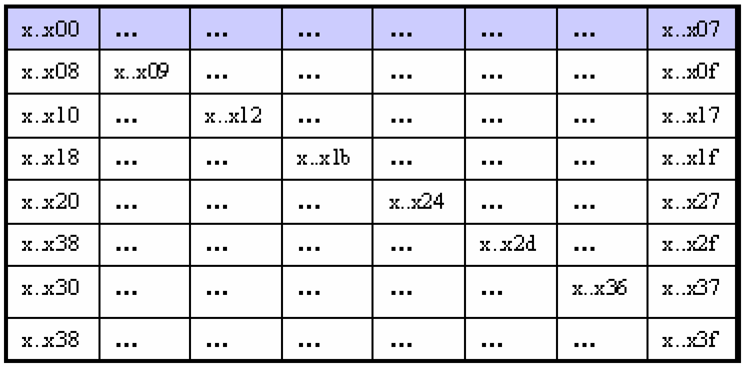
The Symmetrix hardware maintains cache data integrity in 4 KB block increments within a cache slot. This is indicated in Figure 2-1 on page 2-2 by each row of eight 512-byte blocks. Data integrity is maintained using Cyclic Redundancy Check (CRC). The CRC is a very powerful technique to obtain data reliability. The CRC technique is used to protect blocks of data called *frames*. Using this technique, the transmitter appends an extra n- bit sequence to every frame called Frame Check Sequence (FCS). The FCS holds redundant information about the frame that helps the transmitter detect errors in the frame. The technique gained popularity because it combines three advantages:

- Extreme error detection capabilities.
- Little overhead.
- Ease of implementation

CRC comes into play when the I/O is a write operation. The hardware performs the CRC on each 4 KB block whose logical block address (LBA) ends with one of the following hexadecimal values: 00, 08, 10, 18, 20, 28, 30, 38. As a result, the optimum size of an I/O on the Symmetrix array is a multiple of 4 KB where the starting value is lined up on a 4 KB boundary shown in Figure 2-1 on page 2-2.

### 2.1.4.1 Data Alignment

As previously mentioned, the smallest I/O size is 512 bytes. Its address is referred to as the logical block address (LBA) within the SCSI protocol. For example, using this technique, we find that data blocks that are not aligned, are not multiples of eight 512-byte blocks or both create additional work on the part of the storage system. Take the case where we try to write 3k. The CRC must be calculated for the combination of the 3k of new data as well as the remaining 1k of old data in the 4k block. This requires that a read first be performed in order to bring in the old data, overlay the first 3k with the new 3k, re-calculate the CRC on the whole 4k block and writing it out again. We see a 4 KB data blocks considered aligned. In this case, the CRC will be directly calculated by the hardware as this data block is written out to disk.



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| x..x00 | ... | ... | ... | ... | ... | ... | x..x07 |
| x..x08 | x..x09 | ... | ... | ... | ... | ... | x..x0f |
| x..x10 | ... | x..x12 | ... | ... | ... | ... | x..x17 |
| x..x18 | ... | ... | x..x1b | ... | ... | ... | x..x1f |
| x..x20 | ... | ... | ... | x..x24 | ... | ... | x..x27 |
| x..x38 | ... | ... | ... | ... | x..x2d | ... | x..x2f |
| x..x30 | ... | ... | ... | ... | ... | x..x36 | x..x37 |
| x..x38 | ... | ... | ... | ... | ... | ... | x..x3f |

**Figure 2-4**

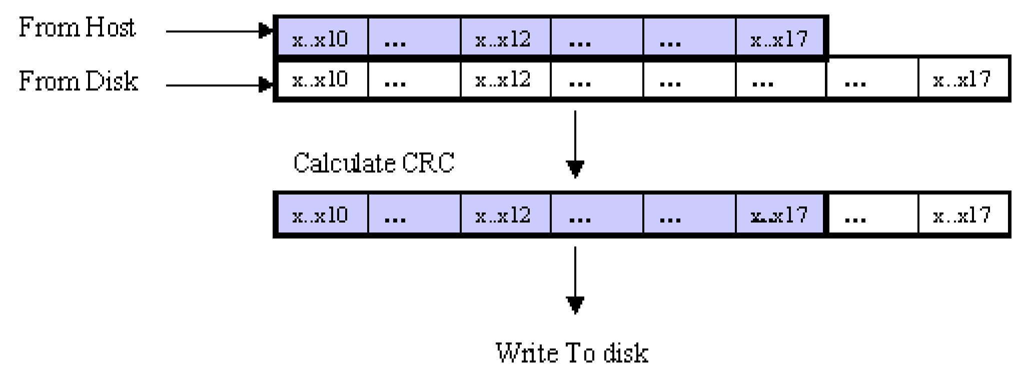


**Figure 2-5**

The result of writing small blocks is as follows:

- 3 KB is written from the host
- 4 KB is read from the disk as a read-modify-write requirement
- 4 KB is written to disk

Consider another situation where we write a 4 KB block that is not aligned on a 4 KB boundary (Figure 2-6).

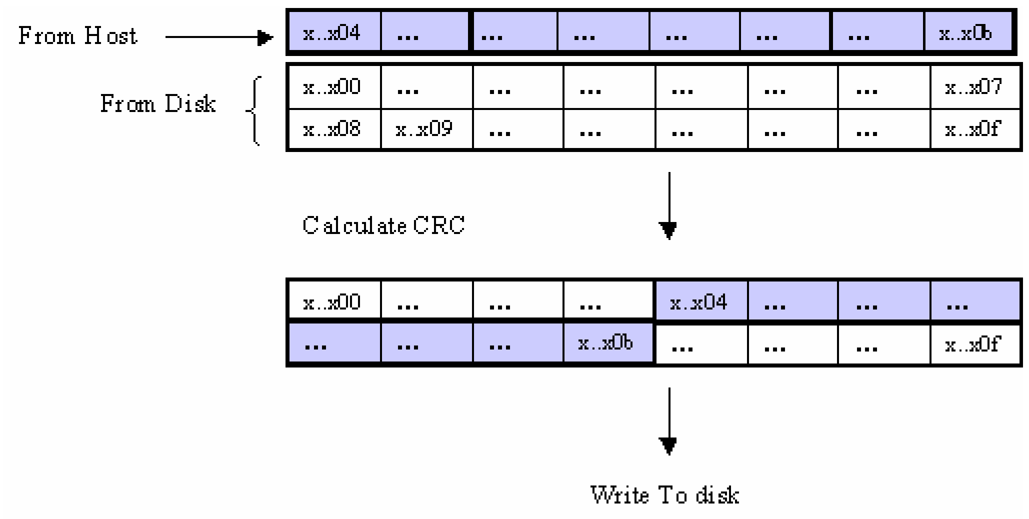


Figure 2-6

In the case of Figure 2-6, we have the following sequence of events:

- The host writes 4 KB to cache
- 8 KB are read from disk as a read-modify-write requirement so that the CRC can be performed on the two adjacent 4 KB blocks
- 8 KB is written to the disk

### 2.1.4.2 Example of the Effect of Unaligned I/O

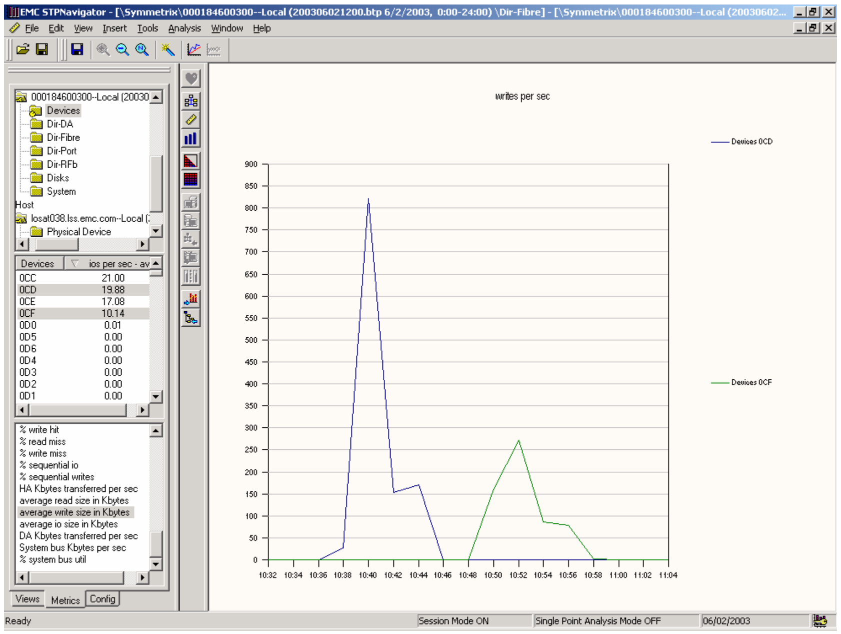
This example is intended to illustrate how the previously discussed topics dealing with alignment and read-modify-write affect I/O response time. Two separate tests were run, both with an I/O size of 32 KB, one was aligned on 32 KB boundaries and the other was not.

### 2.1.4.3 Experiment Specifications

Table 2-1 **Write Rate**

| | Read | Write |
|---|---|---|
| Random/Sequential | | **Random** |
| Sizes (Kbytes) | | 32 |
| Locality of Reference | | 14 GB |
| Thread Count | | 1 |
| Data Protection | | RAID 1 |
| Alignment (yes/no) | | **Yes, no** |

Table 2-1 shows the write rate of each of the devices in the test. The first device, which was aligned, achieved a much higher rate of traffic than the second device. A more detailed analysis follows.


EMC STPNavigator - [\Symmetrix\000184600300--Local (200306021200.btp 6/2/2003, 0:00-24:00) \Dir-Fibre] - [\Symmetrix\000184600300--Local (20030602...

File Edit View Insert Tools Analysis Window Help

000184600300--Local (20030
Devices
Dir-DA
Dir-Fibre
Dir-Port
Dir-RFb
Disks
System
Host
losat038.lss.emc.com--Local (
Physical Device

| Devices | ios per sec - av |
|---|---|
| 0CC | 21.00 |
| 0CD | 19.88 |
| 0CE | 17.08 |
| 0CF | 10.14 |
| 0D0 | 0.01 |
| 0D5 | 0.00 |
| 0D6 | 0.00 |
| 0D4 | 0.00 |
| 0D3 | 0.00 |
| 0D2 | 0.00 |
| 0D1 | 0.00 |

% write hit
% read miss
% write miss
% sequential io
% sequential writes
HA Kbytes transferred per sec
average read size in Kbytes
average write size in Kbytes
average io size in Kbytes
DA Kbytes transferred per sec
System bus Kbytes per sec
% system bus util

Ready | Session Mode ON | Single Point Analysis Mode OFF | 06/02/2003

**Figure 2-7**

For write activities, unaligned I/O can result in one or both of the following situations:

- Allocation of an additional cache slot due to the data addresses extending beyond a single cache slot
- A read-modify-write condition upon destaging the data from the cache to the disk

Either of these conditions will ultimately result in response-time degradation. Therefore, planning for data alignment will definitely guarantee better performance.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| x..x00 | ... | ... | ... | ... | ... | ... | x..x07 |
| x..x08 | x..x09 | ... | ... | ... | ... | ... | x..x0f |
| x..x10 | ... | x..x12 | ... | ... | ... | ... | x..x17 |
| x..x18 | ... | ... | x.x1b | ... | ... | ... | x..x1f |
| x..x20 | ... | ... | ... | x.x24 | ... | ... | x..x27 |
| x..x38 | ... | ... | ... | ... | x..x2d | ... | x..x2f |
| x..x30 | ... | ... | ... | ... | ... | x..x36 | x..x37 |
| x..x38 | ... | ... | ... | ... | ... | ... | x..x3f |
| x..x00 | ... | ... | ... | ... | ... | ... | x..x07 |
| x..x08 | x..x09 | ... | ... | ... | ... | ... | x..x0f |
| x..x10 | ... | x..x12 | ... | ... | ... | ... | x..x17 |
| x..x18 | ... | ... | x.x1b | ... | ... | ... | x..x1f |
| x..x20 | ... | ... | ... | x.x24 | ... | ... | x..x27 |
| x..x38 | ... | ... | ... | ... | x..x2d | ... | x..x2f |
| x..x30 | ... | ... | ... | ... | ... | x..x36 | x..x37 |
| x..x38 | ... | ... | ... | ... | ... | ... | x..x3f |

**Figure 2-8**

Figure 2-8 shows how 32 KB aligned data is positioned in cache slots of size 32 KB. Such alignment is optimal for the Symmetrix since the CRC error protection and detection mechanisms are based optimized without a need to read the disk-based data first.

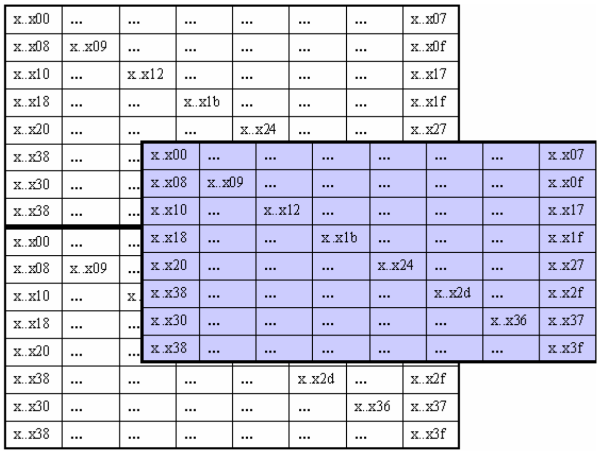
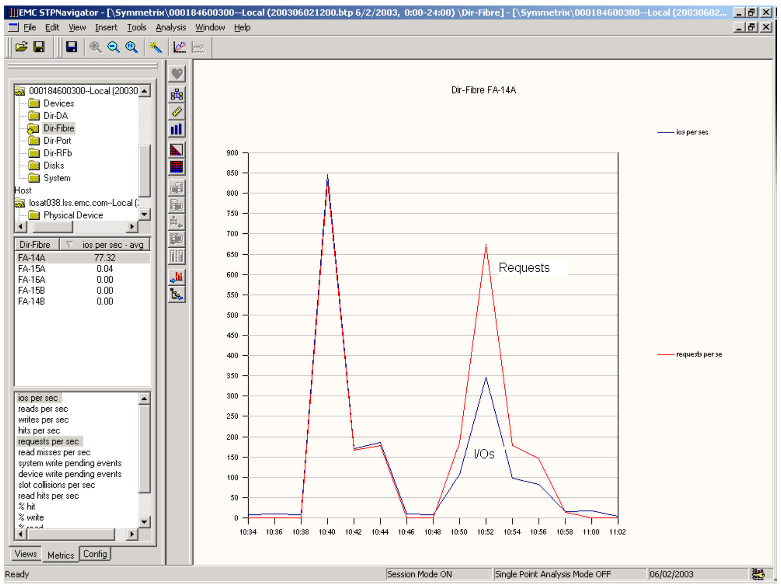

x..x00 ... ... ... ... ... ... x..x07
x..x08 x..x09 ... ... ... ... ... x..x0f
x..x10 ... x..x12 ... ... ... ... x..x17
x..x18 ... ... x..x1b ... ... ... x..x1f
x..x20 ... ... ... x..x24 ... ... x..x27
x..x38 ...
x..x30 ...
x..x38 ...
x..x00 ...
x..x08 x..x09
x..x10 ...
x..x18 ...
x..x20 ...
x..x38 ... ... ... ... x..x2d ... x..x2f
x..x30 ... ... ... ... ... x..x36 x..x37
x..x38 ... ... ... ... ... ... x..x3f
x..x00 ... ... ... ... ... ... x..x07
x..x08 x..x09 ... ... ... ... ... x..x0f
x..x10 ... x..x12 ... ... ... ... x..x17
x..x18 ... ... x..x1b ... ... ... x..x1f
x..x20 ... ... ... x..x24 ... ... x..x27
x..x38 ... ... ... ... x..x2d ... x..x2f
x..x30 ... ... ... ... ... x..x36 x..x37
x..x38 ... ... ... ... ... ... x..x3f

**Figure 2-9**

Figure 2-9 shows a situation where the 32 KB I/O is not aligned, thus requires two separate cache slots.

As discussed in the metrics section earlier, data cache alignment is shown and verified by comparing the directors I/O rate with their request rates. Figure 2-10 shows how in the first test, the I/O rate and request rate are identical, while in the second test, there are approximately twice as many requests as I/Os. This is because the starting address of the I/O block causes the data to span more than one cache slot forcing the director to request two cache slots for every I/O operation.



| Dir-Fibre | ios per sec - avg |
|---|---|
| FA-14A | 77.32 |
| FA-15A | 0.04 |
| FA-16A | 0.00 |
| FA-15B | 0.00 |
| FA-14B | 0.00 |

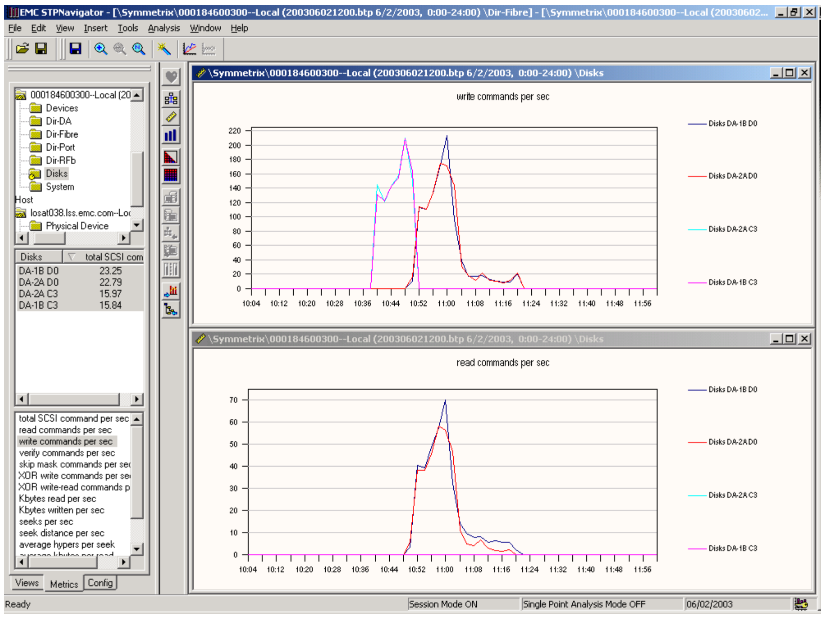
```
ios per sec
reads per sec
writes per sec
hits per sec
requests per sec
read misses per sec
system write pending events
device write pending events
slot collisions per sec
read hits per sec
% hit
% write
% read
```

**Figure 2-10 I/Os vs. Requests**

### 2.1.4.4 Read-Modify-Write

In order to show the effect of misalignment, Figure 2-11 shows the disk activity associated with the read-modify-write. In the upper graph, there are two sets of two active disks executing write activity (this is a RAID 1 or mirrored environment). The first set is associated with the device that was writing aligned data and therefore the lower graph shows no read activity on these disks.

For the second set of active disks in the upper graph, which shows the write activity for the unaligned device, there is notable read activity shown in the lower graph.


EMC STPNavigator - [\Symmetrix\000184600300--Local (200306021200.btp 6/2/2003, 0:00-24:00) \Dir-Fibre] - [\Symmetrix\000184600300--Local (20030602...
File Edit View Insert Tools Analysis Window Help
000184600300--Local (20
Devices
Dir-DA
Dir-Fibre
Dir-Port
Dir-RFb
Disks
System
Host
losat038.lss.emc.com--Loc
Physical Device
\Symmetrix\000184600300--Local (200306021200.btp 6/2/2003, 0:00-24:00) \Disks
write commands per sec
Disks DA-1B D0
Disks DA-2A D0
Disks DA-2A C3
Disks DA-1B C3
read commands per sec
total SCSI command per sec
read commands per sec
write commands per sec
verify commands per sec
skip mask commands per sec
XOR write commands per se
XOR write-read commands p
Kbytes read per sec
Kbytes written per sec
seeks per sec
seek distance per sec
average hypers per seek
Views
Metrics
Config
Ready
Session Mode ON
Single Point Analysis Mode OFF
06/02/2003

| Disks | total SCSI com |
|---|---|
| DA-1B D0 | 23.25 |
| DA-2A D0 | 22.79 |
| DA-2A C3 | 15.97 |
| DA-1B C3 | 15.84 |

Figure 2-11 **Disk Activity Associated with the Read-Modify-Write**

## Response Times

The most obvious effect of misalignment is in the response time. Keep in mind that the response time is the most critical metric of performance. It is here that we usually look first.

The view in Figure 2-12 is presented from both the host and the storage-system perspective. The intent is to verify that both are measuring the same values. This is the trivial case where there is only one thread active with no queues anywhere. In this case, it is fully that both host and storage present the same response times.

The important information relative to this experiment is that in the misaligned case, the response time is almost twice that of the aligned case. The additional time is due to the additional work that the front-end director has to do in order to allocate two cache slots instead of one. Also, an additional disk read is required by the back-end director to ensure that the correct CRC is calculated prior to writing the data to the disk.

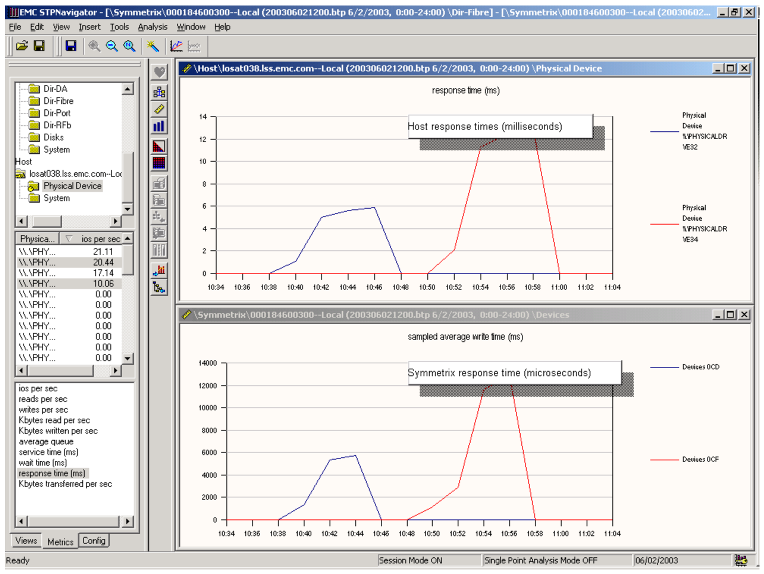
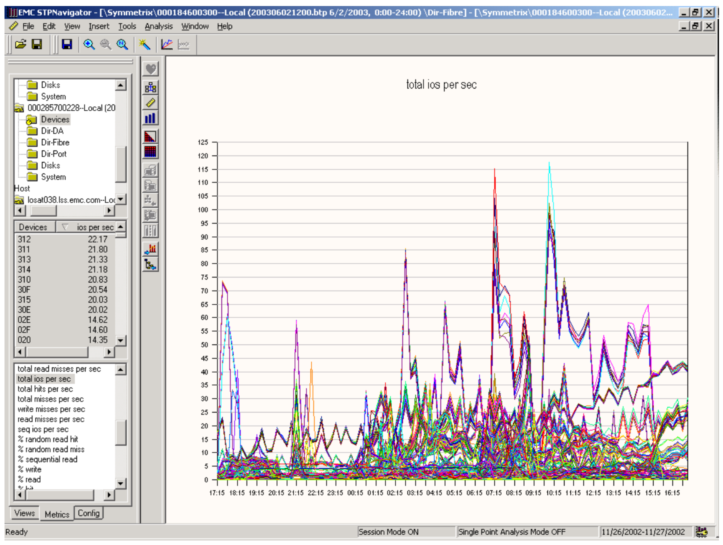

EMC STPNavigator - [\Symmetrix\000184600300--Local (200306021200.btp 6/2/2003, 0:00-24:00) \Dir-Fibre] - [\Symmetrix\000184600300--Local (20030602...
File Edit View Insert Tools Analysis Window Help
\Host\losat038.lss.emc.com--Local (200306021200.btp 6/2/2003, 0:00-24:00) \Physical Device
response time (ms)
Host response times (milliseconds)
Physical Device \\.\PHYSICALDRIVE32
Physical Device \\.\PHYSICALDRIVE34
\Symmetrix\000184600300--Local (200306021200.btp 6/2/2003, 0:00-24:00) \Devices
sampled average write time (ms)
Symmetrix response time (microseconds)
Devices 0CD
Devices 0CF
Dir-DA
Dir-Fibre
Dir-Port
Dir-RFb
Disks
System
Host
losat038.lss.emc.com-Loc
Physical Device
System
ios per sec
reads per sec
writes per sec
Kbytes read per sec
Kbytes written per sec
average queue
service time (ms)
wait time (ms)
response time (ms)
Kbytes transferred per sec
Views
Metrics
Config
Ready
Session Mode ON
Single Point Analysis Mode OFF
06/02/2003

Figure 2-12

## 2.1.5 Write-Pending Limits

The storage-system cache is a set of memory boards of a fixed size at any point in time. The size of the cache can vary depending on the number of logical devices and physical components that compose the system. These require a fixed amount of memory for internal management purposes and the remaining space is used for user data. The cache slots are a logical partitioning of the user-data cache into a set of manageable pieces that are placed in one or more LRU tables and are provisioned out as needed. Data in the slots ages over time and is either written over or de-staged to the disks.

The cache serves all logical units (LUNs in open systems) in the storage system. Activity at the cache can be quite chaotic at times. For instance, Figure 2-13 on page 2-12 shows the total I/O activity for one Symmetrix system. There are 1023 logical devices that generate the activity in Figure 2-13. All of these devices compete for cache resources causing staging and destaging of data in and out of the cache.


EMC STPNavigator - [\Symmetrix\000184600300--Local (200306021200.btp 6/2/2003, 0:00-24:00) \Dir-Fibre] - [\Symmetrix\000184600300--Local (20030602...

| Devices | ios per sec |
|---|---|
| 312 | 22.17 |
| 311 | 21.80 |
| 313 | 21.33 |
| 314 | 21.18 |
| 310 | 20.83 |
| 30F | 20.54 |
| 315 | 20.03 |
| 30E | 20.02 |
| 02E | 14.62 |
| 02F | 14.60 |
| 020 | 14.35 |

total read misses per sec
total ios per sec
total hits per sec
total misses per sec
write misses per sec
read misses per sec
seq ios per sec
% random read hit
% random read miss
% sequential read
% write
% read

Session Mode ON
Single Point Analysis Mode OFF
11/26/2002-11/27/2002

**Figure 2-13**

It is conceivable that on occasion one or more devices may begin to issue continuous writes for a sustained period of time causing the data to flood all available cache. Under these circumstances all other logical device activity is blocked due to lack of cache resources.

There are various techniques used by the industry to minimize the effect of such events. The method that the Symmetrix system uses is by a defining a series of thresholds that prevent any one device from blocking other devices from access to the cache.

There are three different types of thresholds defined (see Figure 2-14 on page 2-13):

- System Write Pending Limit
- Device Write Pending Limit
- DA Write Pending Limit

Together they maintain the flow of data through the system by limiting the impact of write-intensive operations on throughput and response times.

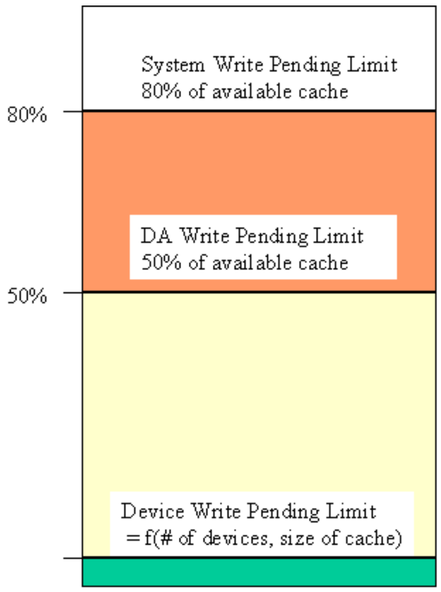

System Write Pending Limit
80% of available cache
80%
DA Write Pending Limit
50% of available cache
50%
Device Write Pending Limit
= f(# of devices, size of cache)

**Figure 2-14**

### 2.1.5.1 System Write-Pending Limit

To ensure that some read activity is able to continue even if many devices are busy writing, the Symmetrix system defines a global system write-pending threshold set at the 80 percent of user cache. To reach that level, multiple devices must be active, simultaneously writing cache.

When the system write-pending limit is reached, all writes begin to be throttled and are counted as "misses." They now wait until space is made available in cache by destaging to disk.

### 2.1.5.2 Device Write-Pending Limit

A write-pending cache slot contains written data that has not yet been destaged to the disk. A cache slot with data that has not been destaged cannot be reused by anyone other than an I/O that is rewriting or reading from that same device address. The Symmetrix system maintains a count per device of the number of slots that are write pending and continuously checks that number against the predefined threshold.

A device write-pending limit is assigned to each device. The limit value is variable from system to system and is a function of the total cache size and the number of devices configured in the system. The purpose of this threshold is to limit the effect that one device has on the available cache for all other devices. The result is that once the device write activity reaches its write-pending threshold, the device must wait until some data is destaged to disk before continuing to write to cache.

The Symmetrix system maintains two different values for device write-pending limits:

- **Base level** — Established relative to the previously mentioned parameters
- **Maximum limit** — Three times the base level

The two values have different purposes.

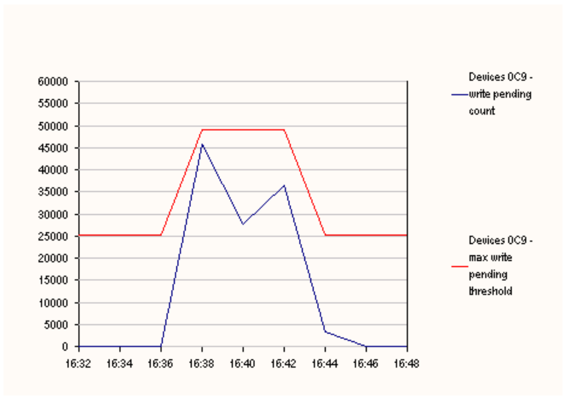
#### Base Level

The base level senses that the device is in a continuous write mode and has the potential to use up more cache. When this level is reached, the Symmetrix system goes into a mode referred to as *aggressive write* mode. Up until that point, the write-to-disk activity with respect to that device is maintaining a lower priority to read activity. However, once the base write-pending limit has been reached, the priority for writes to the disk has increased to the same level as the reads.

### Maximum Limit

The purpose of the maximum write-pending limit (three times the base), is to actually halt all new writes until space is made available through destaging. The result of this limit is that the I/O write rate for that device is decreased to the effective rate of the destaging activity. This is a function of how fast the disk can react to that requirement. The ability of the disk to effectively respond to the destaging activity is a function of what other devices are active on that disk from overall system traffic.

Figure 2-15 shows a case where the writes to a device exceed the base write pending limit of 25000 cache slots. However, since we haven't reached the maximum level yet (three times the base), the write rate has not changed substantially.



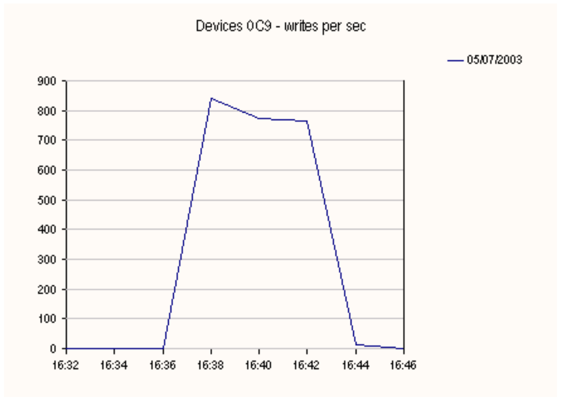
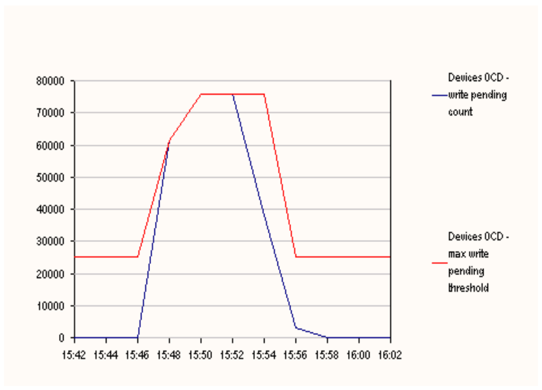


**Figure 2-15**

In the case of the activity shown in Figure 2-16 on page 2-15, we see the threshold line is reaching the maximum limit of 75000 (3 x 25000) and is sustained at that level. The result of this is two fold:

- The destaging activity goes on, the write rate begins to decrease and approaches the speed of the disk's ability to write the data
- The write activity begins to experience cache "misses," that is, it is not able to find places in the cache to write until destaging occurs (Figure 2-17 on page 2-15)



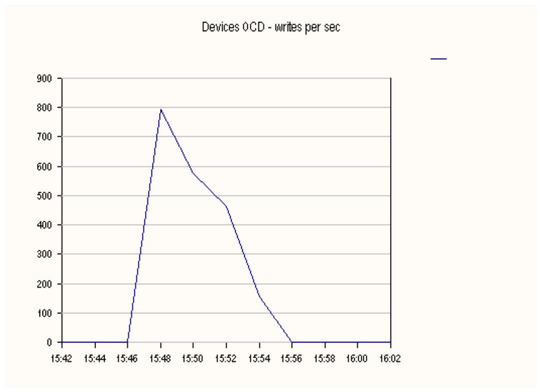


**Figure 2-16**

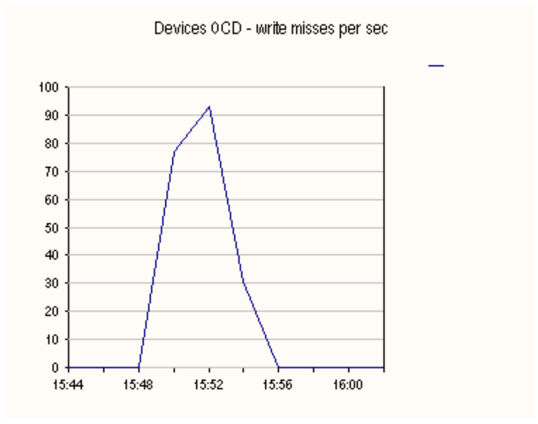


**Figure 2-17**

Normally, such occurrences in a system with many devices are limited to only a handful. Under such circumstances, these devices experience read-misses, which translate into longer response times. The solution to such a problem is to partition these devices into a set of metadevices where each member of the meta has its own write pending limit. The tradeoff is that as more devices are introduced into the system, the write-pending limit for each device becomes smaller, but unless one had to substantially increase the overall number of devices, this change would generally be manageable.

### 2.1.5.3 DA Write Pending Limit

When the cache is 50 percent filled with write-pending slots, the disk directors raise the priority and accelerate the rate of the destaging tasks. This is necessary because under unusual conditions it is possible for this limit to be reached without any specific device reaching its own write-pending limit yet, in aggregate, the sum for all devices reaches the 50 percent level.

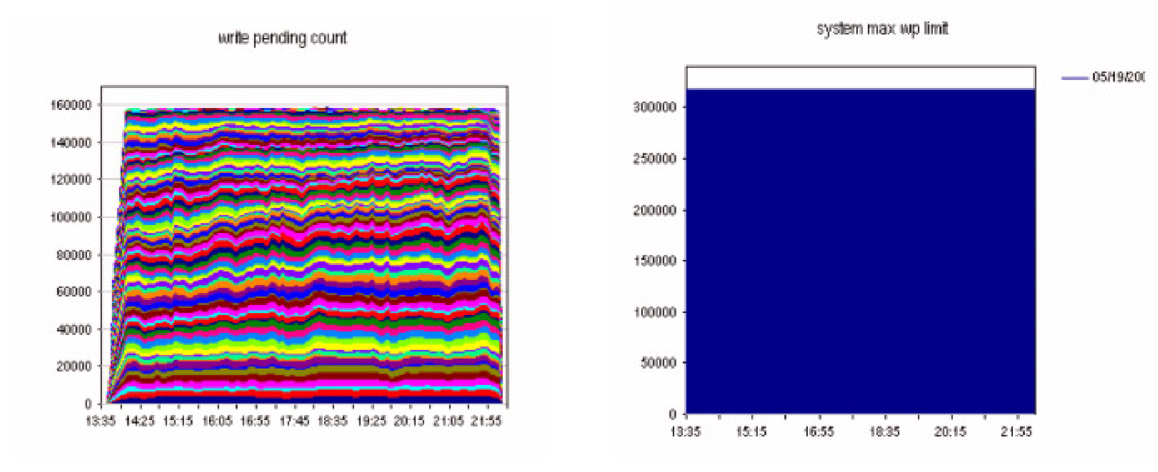


Figure 2-18

In the above figure, notice the condition that triggers the DA Write Pending Limit. The sum total of the write pending slots by all the devices is one half that of the System Max Write Pending Limit. However, at no time do any of the devices reach their own Device Write Pending Limit.

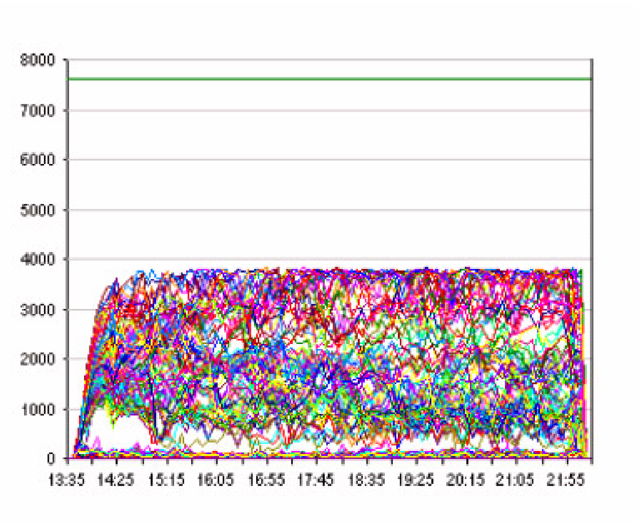


Figure 2-19

# 3
# Analyzing Symmetrix Performance

This chapter reviews these topics:

## 3.1 Overview

Consider the value of a storage system. Its main objective is to provide multiple paths to a large storage pool via a high-speed cache. The ultimate goals are centralized storage management and speed of access. There are several such systems on the market today from such companies as HDS, IBM, and HP, as well as multiple systems from EMC. There are many similarities in architecture among these systems.

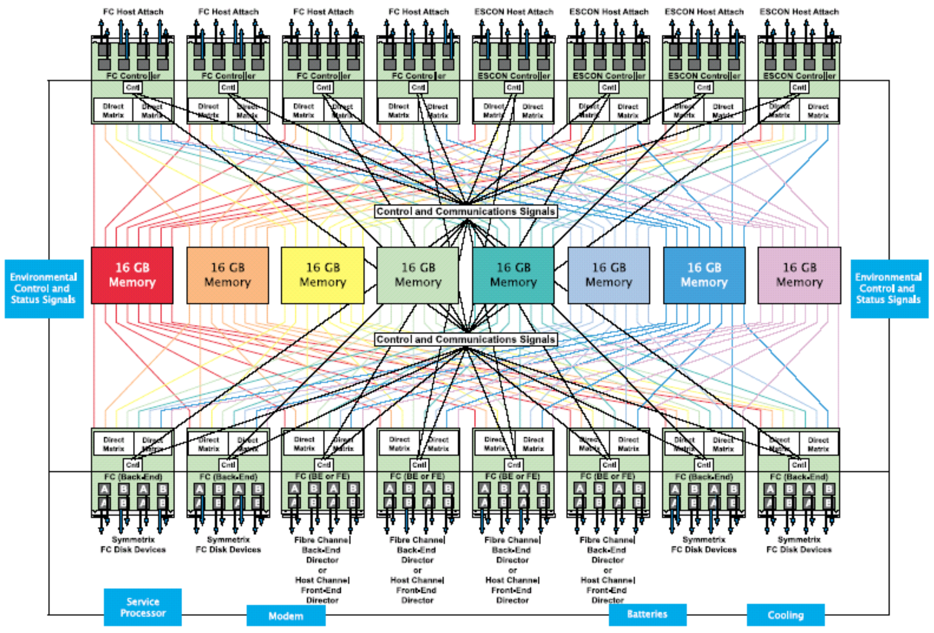

FC Host Attach
ESCON Host Attach
FC Controller
ESCON Controller
Cntl
Direct Matrix
Control and Communications Signals
16 GB Memory
Environmental Control and Status Signals
FC (Back-End)
FC (BE or FE)
Symmetrix FC Disk Devices
Fibre Channel Back-End Director or Host Channel Front-End Director
Service Processor
Modem
Batteries
Cooling

**Figure 3-1 The DMX2000 Configuration**

The key elements of a storage system are centered around the cache. Such systems are also referred to as dual-stage systems, where data is first placed in the cache for both input and output operations. As a result, there are two key areas for optimization for such a system. The first is in the balancing of the workload across the front-end ports and across the back-end disks. The second is in the algorithms that cleverly perform the staging and destaging of the data to/from the cache. The best place to measure the effectiveness of workload balance and algorithms is in the I/O response time as seen from the host perspective.

The latest version of ControlCenter Workload Analyzer includes the response time metric for Symmetrix devices. Devices represent the logical work units to which I/Os are performed. Symmetrix devices are mapped, directly and indirectly, to host devices, and host applications generate I/Os to these host devices, again, directly and indirectly.

The term *indirect* here addresses the general subject of virtualization, where an application file represents a database table, which in turn is mapped to a volume manager volume that may be directly associated with a host operating system device. That device is then mapped to the Symmetrix device, which represents a striped set of multiple devices and ultimately resides on multiple disk drives (Figure 3-2 on page 3-3).

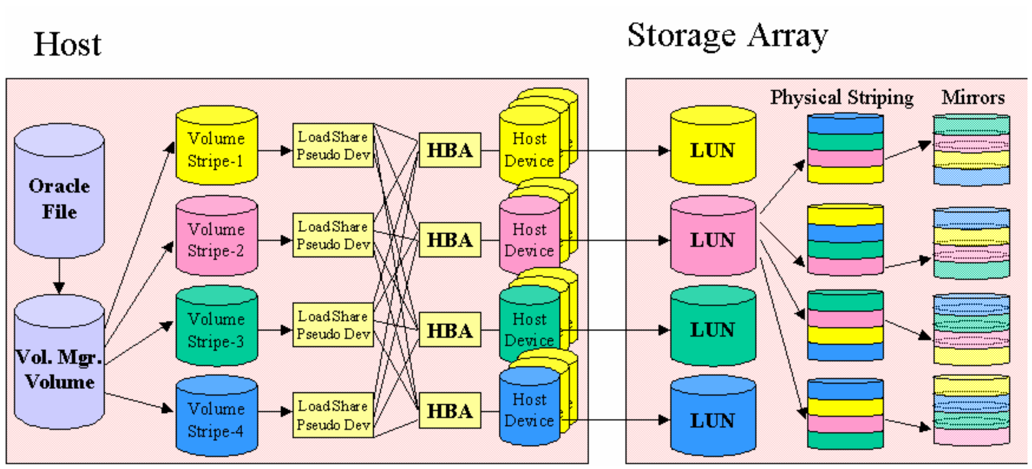


Figure 3-2

The introduction of the response metric provides the performance analyst with the means to isolate the direct contribution to the performance by the storage system with respect to the hosts and applications connected to that system.   Response time is the best measurement of service quality provided by any system. Understanding how to determine where time is spent in the system provides the best means for problem identification, bottleneck detection, system tuning, and capacity planning.

## 3.2 I/O Types

The lifecycle of six I/O types will be explored. They are write-hits, read-hits, write-misses, read-misses, sequential writes, and sequential reads.

The nature of I/O hit activity will show how the workload is characterized on the front end, allowing you to determine the load level at each of the front-end ports and directors. This understanding will assist you in properly balancing the workload on the storage system as well as conduct capacity planning for the system.

I/O miss activity will show you how the back end participates in the performance of the I/O. The back end also plays a critical role in the success of sequential I/O activity by managing the rapid destaging of data from the cache to the disks due to write activity, as well as prefetching and staging data for sequential read activity, elevating most sequential read I/Os to read-hits.

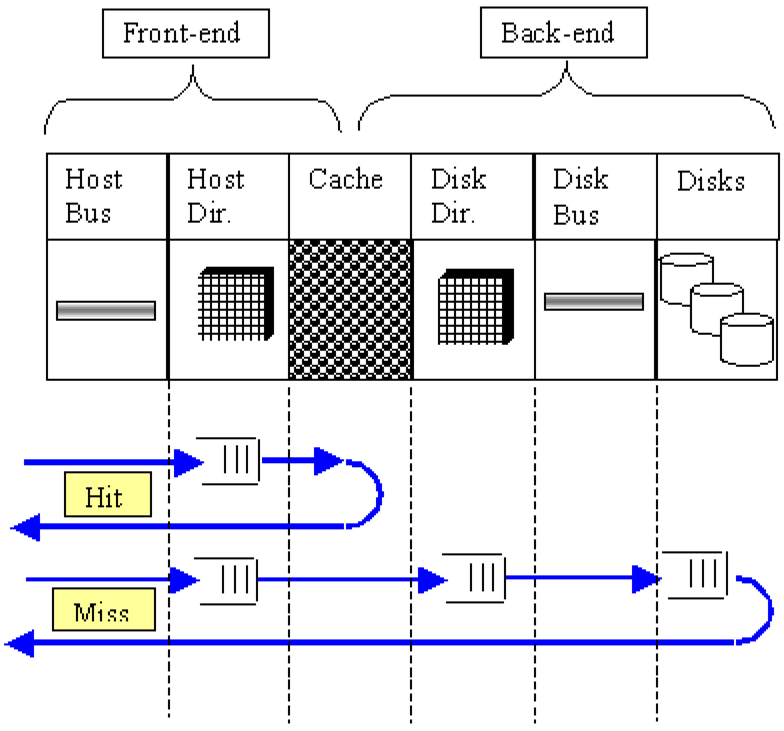


**Figure 3-3**

Figure 3-3 shows the path of the hit and miss operations through the major components of the Symmetrix array. The response times for the I/Os are measured and reported by the host director and reflect the time from the receipt of the I/O command from the host to the completion and receipt of an acknowledgement by the host.

WLA lets the analyst view the I/O traffic at the storage array with substantial detail. There are metrics that describe the nature of the I/O to each device in the system. From this data, you can determine the following information about each device:

- Overall I/O rate
- Percentage of reads and writes
- Percentage of read hits

- Percentage of sequential reads (mainframe systems only)
- Percentage of write hits
- Percentage of sequential writes (mainframe systems only)
- Average read I/O size
- Average write I/O size
- Average (sampled) read response times (open systems only)
- Average (sampled) write response times (open systems only)

The response time of an I/O operation is a function of the following factors:

- Read or write
- % hit (which may reflect the effect of sequential activity)
- I/O size (which affects alignment)
- The number of threads active at the host generating asynchronous I/O activity

A combination of proper system configuration and software algorithms is used to optimize the I/O response time through the Symmetrix array. The configuration techniques rely on multiple load balancing methods. There are automated tools such as EMC PowerPath®, a host-based port load-balancing package, and the Symmetrix Optimizer, which load balances the disks on the back end. There are also manual zoning procedures that route traffic from hosts through switches to the storage systems as well as manual methods for moving devices from disk to disk.

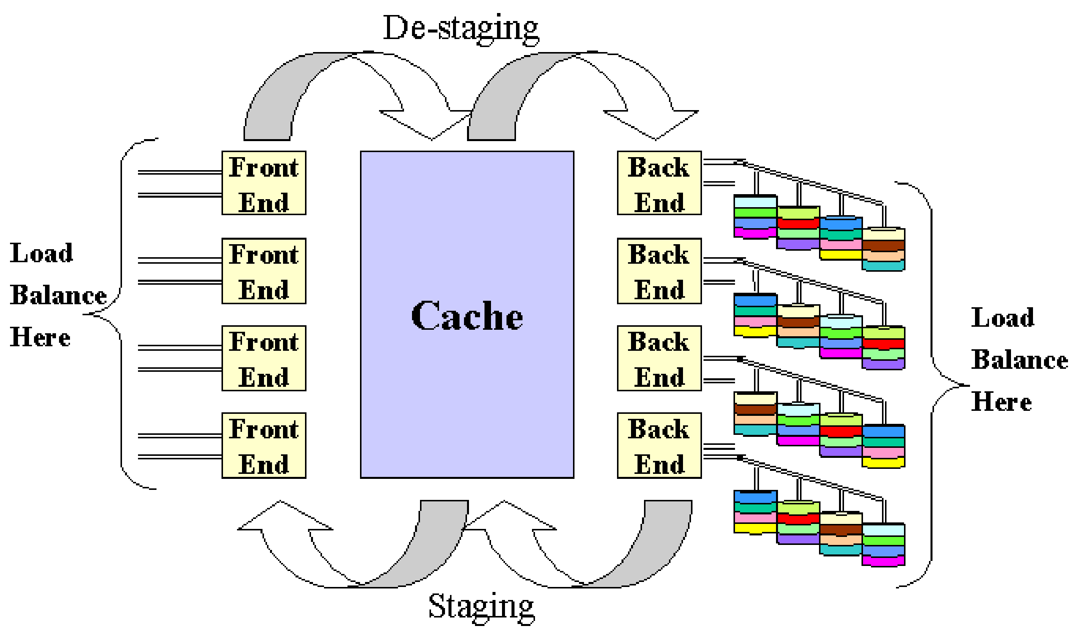


Figure 3-4

## 3.3 The Front End

Performance metrics are available for the Symmetrix front-end components independently for ports and host directors. However, these components work in tandem when an I/O occurs.

Consider an I/O that is a write-hit. Figure 3-5 shows that a write-hit operation travels from the host bus to the host director and to the cache. The completion time of such an I/O is the sum of the service times at the bus and at the director CPU. The work done by the CPU involves the finding and allocation of the cache slot where the write-data will reside until it is destaged to the disk at some later time. Since the response time as seen by the host does not include in it the time to destage the data to the disk, this discussion will ignore that activity for now.

### 3.3.1 Fibre Bus Service Times

The speed of a 1 Gb Fibre bus is 132,750,000 bytes per second.

The data is transmitted in a series of frames following a specific protocol that requires additional addressing information for each frame. Figure 3-5 shows that each packet is added 100 bytes for addressing.

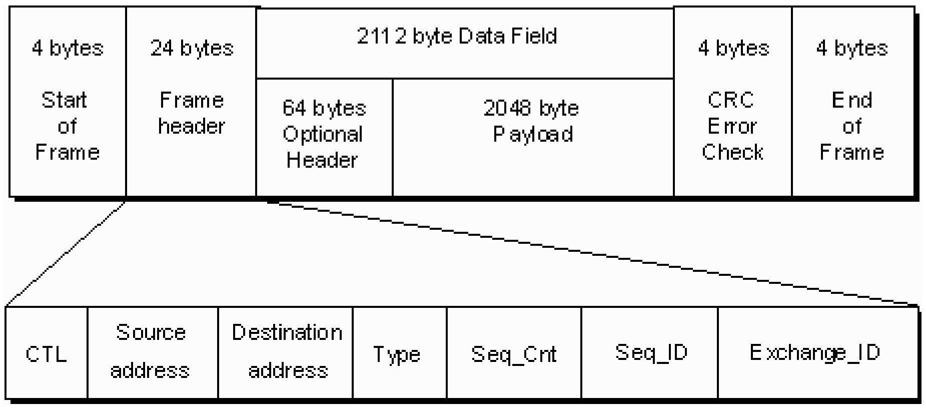


**Figure 3-5**

Since each packet contains no more than 2112 data bytes plus 100 protocol bytes, it is easy to calculate the time that any given I/O spends on the bus:

The formula:

**Data size of whole packets:** INT (I/O size/2112)*2212

+

**The remaining data:** MOD (I/O size,2112)+100

The sum of the two divided by 132,750,000 to calculate fraction-of-second use.

Multiply by 1000 to convert to milliseconds.

| I/O Packet Size | Time on 1 Gb Fibre (in ms) |
|---:|---:|
| 512 | 0.005 |
| 1024 | 0.008 |
| 2048 | 0.016 |
| 4096 | 0.032 |
| 8192 | 0.062 |
| 16384 | 0.124 |
| 32768 | 0.248 |
| 65536 | 0.494 |
| 131072 | 0.988 |
| 262144 | 1.975 |
| 524288 | 3.950 |
| 1048576 | 7.900 |

## 3.3.2 The Front-End Model

In a Synchronous mode of operation where a single thread is issuing I/Os, a subsequent I/O appears only after the completion of the currently active I/O. The I/O operation consists of the time at the bus and the time at the director. Since the next I/O does not appear begin until the current one completes, and since for a given I/O size we know the time that it spends on the bus, if we knew how long the I/O spent at the director CPU, we could estimate the maximum capabilities of the front end for write-hits of that I/O size.

An example is provided in the following table. One I/O size is selected for which the service time on the Fibre bus was calculated earlier. In the example, a range of service times at the director were provided in order to observe the relationship among the two components.

| I/O Packet Size | Service time on 1 Gb Fibre (in ms) | Service Time at the Host Director (in ms) | Total Service Time | Total Number of I/Os per Second | % Time Bus Busy | % Time Director Busy |
|---:|---:|---:|---:|---:|---:|---:|
| 32768 | 0.25 | 0.1 | 0.35 | 2877 | 0.71 | 0.29 |
| | | 0.2 | 0.45 | 2234 | 0.55 | 0.45 |
| | | 0.3 | 0.55 | 1826 | 0.45 | 0.55 |
| | | 0.4 | 0.65 | 1544 | 0.38 | 0.62 |
| | | 0.5 | 0.75 | 1338 | 0.33 | 0.67 |

Total service time is the sum of the time at the Fibre bus and at the host director.

The total number of I/Os per second was calculated by dividing 1 second (1000 milliseconds) by the total service time.

Percent-busy time was calculated by multiplying the individual service times by the calculated I/O rate and dividing the value by the elapsed time.

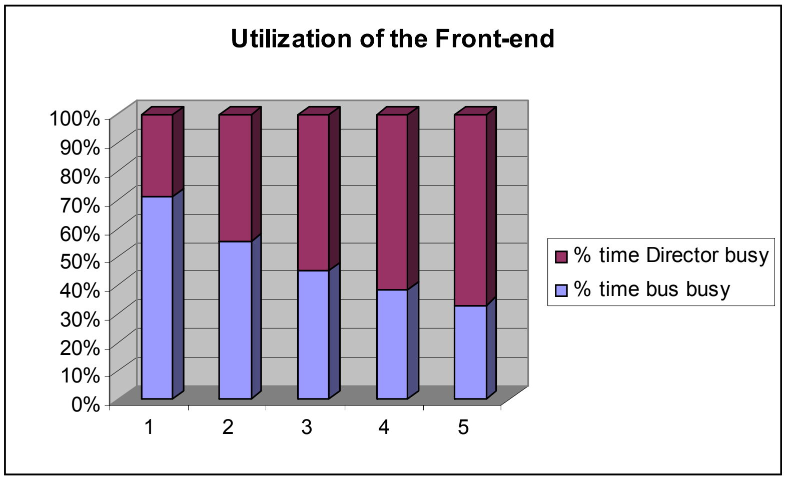


**Figure 3-6**

Figure 3-6 plots the cumulative utilization of the port and the director. Together these two components compose the front end. We notice the following observations:

- 100 percent of the time is accounted for by the sum of the two components. An I/O operation is either busy at the bus or at the CPU.
- In the single-threaded Synchronous mode, no single component can reach 100 percent busy. The two units work as a single machine.

### 3.3.3 Random Write Hits

#### 3.3.3.1 Single Threaded Synchronous Write-Hit

Experiments are run in order to see if we can calculate the host director service times for a write-hit operation. The experiments will show data collected from both the host and the Symmetrix array. We will show how the data is correlated between the two systems and how the response time for the I/O is decomposed. The data from the experiments will then be entered into the model in table in section *3.3.2 The Front-End Model* on page 3-7 to check if the experiments confirm the calculations.

This chapter discusses how cache is managed in the Symmetrix array. A specifically applicable discussion addresses the issue around data alignment within the cache slots. With the availability of response-time measurements in WLA, we are now able to estimate the costs of misalignment.

A data buffer is identified as unaligned by the host director as it tries to assign a cache slot to the data. Costly misalignment means that a data block that can fit within a single cache slot actually requires more than one slot. CPU time is needed to acquire each cache slot. It involves locking and updating the LRU tables and assigning the slots to the I/O.

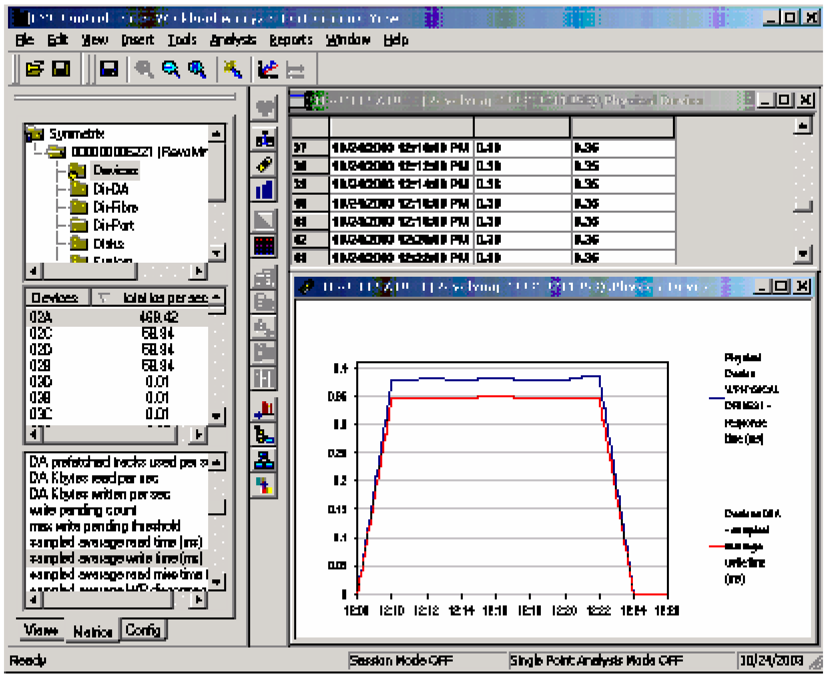
**8 KB Random Writes, Aligned**

We begin with the response times as viewed by both the host and the Symmetrix system for the device under test. Figure 3-7 shows that the device is experiencing the following response times:

**Host:** .38 milliseconds

**Symmetrix:** .35 milliseconds

The 30-microseconds difference is simply the overhead at the host associated with processing and recording the I/O.



| | | | |
|---|---|---|---|
| 37 | 10/24/2003 12:1[illegible] PM | 0.40 | 0.36 |
| 38 | 10/24/2003 12:1[illegible] PM | 0.38 | 0.35 |
| 39 | 10/24/2003 12:1[illegible] PM | 0.38 | 0.35 |
| 40 | 10/24/2003 12:1[illegible] PM | 0.38 | 0.35 |
| 41 | 10/24/2003 12:1[illegible] PM | 0.38 | 0.35 |
| 42 | 10/24/2003 12:2[illegible] PM | 0.38 | 0.36 |
| 43 | 10/24/2003 12:2[illegible] PM | 0.40 | 0.36 |

| Devices | total ios per sec |
|---|---|
| 02A | 468.42 |
| 02C | 58.94 |
| 02D | 58.94 |
| 02B | 58.94 |
| 03D | 0.01 |
| 03B | 0.01 |
| 03C | 0.01 |

DA prefetched tracks used per s
DA Kbytes read per sec
DA Kbytes written per sec
write pending count
max write pending threshold
sampled average read time (ms)
sampled average write time (ms)
sampled average read miss time (

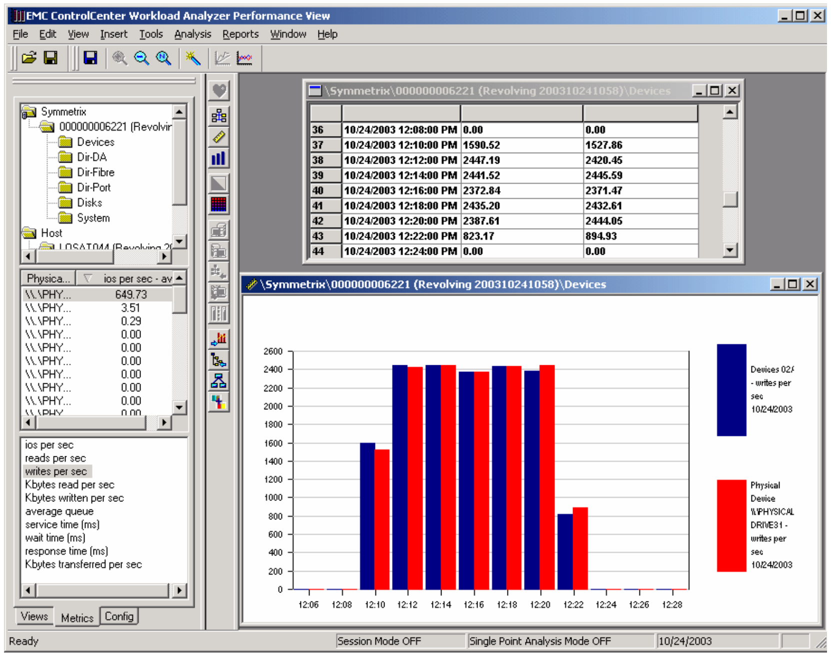
Ready | Session Mode OFF | Single Point Analysis Mode OFF | 10/24/2003

Figure 3-7

Using the .38 milliseconds as the full elapsed time per I/O for the thread initiated at the host, we can calculate the total number of I/Os that this test can generate:

1000 milliseconds (1 second) divided by .38 milliseconds per I/O = 2631.6 I/Os

We then proceed to view the data with WLA and we see the following (Figure 3-8 on page 3-10).


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221 (Revolvir
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 2(
Views
Metrics
Config
\Symmetrix\000000006221 (Revolving 200310241058)\Devices
Devices 02A - writes per sec 10/24/2003
Physical Device \\.\PHYSICAL DRIVE31 - writes per sec 10/24/2003
Ready
Session Mode OFF
Single Point Analysis Mode OFF
10/24/2003

| Physica... | ios per sec - av |
|---|---|
| \\.\PHY... | 649.73 |
| \\.\PHY... | 3.51 |
| \\.\PHY... | 0.29 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |

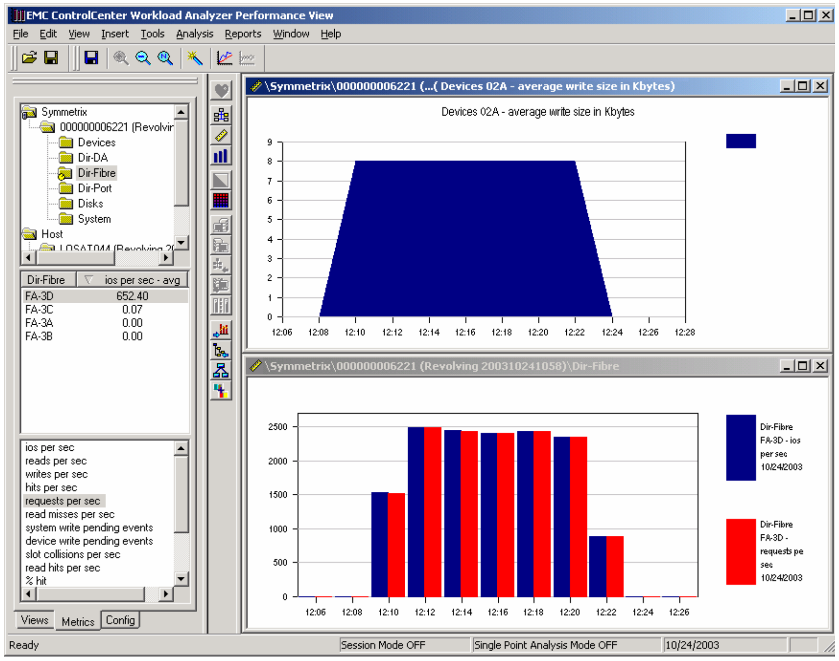
ios per sec
reads per sec
writes per sec
Kbytes read per sec
Kbytes written per sec
average queue
service time (ms)
wait time (ms)
response time (ms)
Kbytes transferred per sec

| | | | |
|---|---|---|---|
| 36 | 10/24/2003 12:08:00 PM | 0.00 | 0.00 |
| 37 | 10/24/2003 12:10:00 PM | 1590.52 | 1527.86 |
| 38 | 10/24/2003 12:12:00 PM | 2447.19 | 2420.45 |
| 39 | 10/24/2003 12:14:00 PM | 2441.52 | 2445.59 |
| 40 | 10/24/2003 12:16:00 PM | 2372.84 | 2371.47 |
| 41 | 10/24/2003 12:18:00 PM | 2435.20 | 2432.61 |
| 42 | 10/24/2003 12:20:00 PM | 2387.61 | 2444.05 |
| 43 | 10/24/2003 12:22:00 PM | 823.17 | 894.93 |
| 44 | 10/24/2003 12:24:00 PM | 0.00 | 0.00 |

**Figure 3-8**

Figure 3-8 shows that the actual achieved I/O rate was approximately 2400 I/Os per second. This translates to elapsed time per I/O of .416 milliseconds. The reason for this discrepancy is that the additional 36 microseconds is the time needed by the load-generating application to process the completion of one I/O and send out the next.

Before we do the calculations for the model, we first verify that the data is 8 KB in size and aligned.


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221 (Revolvir
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 2(
Views
Metrics
Config
\Symmetrix\000000006221 (...( Devices 02A - average write size in Kbytes)
Devices 02A - average write size in Kbytes
\Symmetrix\000000006221 (Revolving 200310241058)\Dir-Fibre
Dir-Fibre FA-3D - ios per sec 10/24/2003
Dir-Fibre FA-3D - requests pe sec 10/24/2003
Ready
Session Mode OFF
Single Point Analysis Mode OFF
10/24/2003

| Dir-Fibre | ios per sec - avg |
|---|---|
| FA-3D | 652.40 |
| FA-3C | 0.07 |
| FA-3A | 0.00 |
| FA-3B | 0.00 |

ios per sec
reads per sec
writes per sec
hits per sec
requests per sec
read misses per sec
system write pending events
device write pending events
slot collisions per sec
read hits per sec
% hit

**Figure 3-9**

The devices contain the metric ‘average write size in kbytes. The top view in Figure 3-9 verifies that the test was an 8 KB write.

The directors statistics indicate alignment since it is the directors that need to read the LRU tables, request cache slots, and assign them to the I/O data. The lower view shows that the I/O rate matches the request rate, thus confirming that the cache slot to I/O ratio is one-to-one and thus, aligned.

With this test, we can now build a table that shows how response time is distributed along the path of the I/O as well as estimate the time that the host director spends in allocating the cache slot and acknowledging the I/O completion to the host.

| | | Measured Response Time (in ms) | Layer Service Time (in ms) |
|---|---|---|---|
| Measured | Application Time | 0.416 | 0.036 |
| | Host time | 0.38 | 0.03 |
| | Symmetrix | 0.35 | |
| Calculated | Bus (Figure 6) | 0.062 | 0.062 |
| | Host director | 0.288 | 0.288 |

With the data collected from WLA and host director time calculated for the aligned 8 KB I/O, we can now build a new model to show what the actual utilization levels for the bus and director were for the achieved load.

| I/O Packet Size | Service Time on 1 Gb Fibre (in ms) | Service Time at the Host Director (in ms) | Total Service Time | Total Number of I/Os per Second | % Tme Bus Busy | % Time Director Busy |
|---|---|---|---|---|---|---|
| 8192 | 0.06 | 0.288 | 0.35 | 2857 | 17.71 | 82.29 |
| | | 0.288 | 0.35 | 2400 | 14.88 | 69.12 |

The first line uses the calculated service time for the host director to estimate the maximum possible I/O rate of the front end, provided there were no delays at the host or the application.

The second line calculates the actually achieved utilization levels of both host bus and host director based on the achieved I/O rate of 2400 I/Os per second.

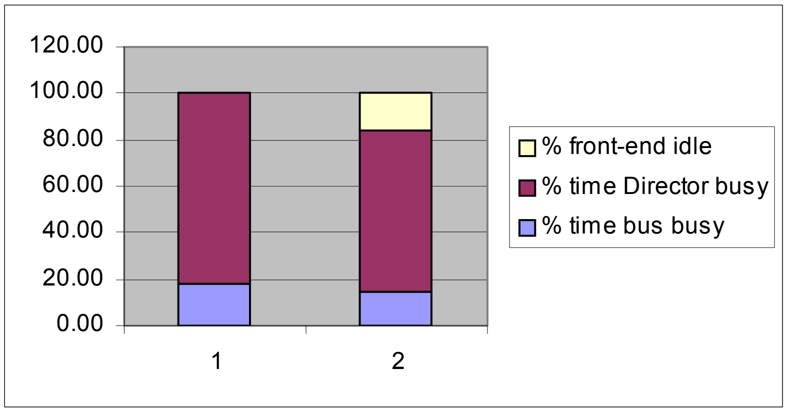


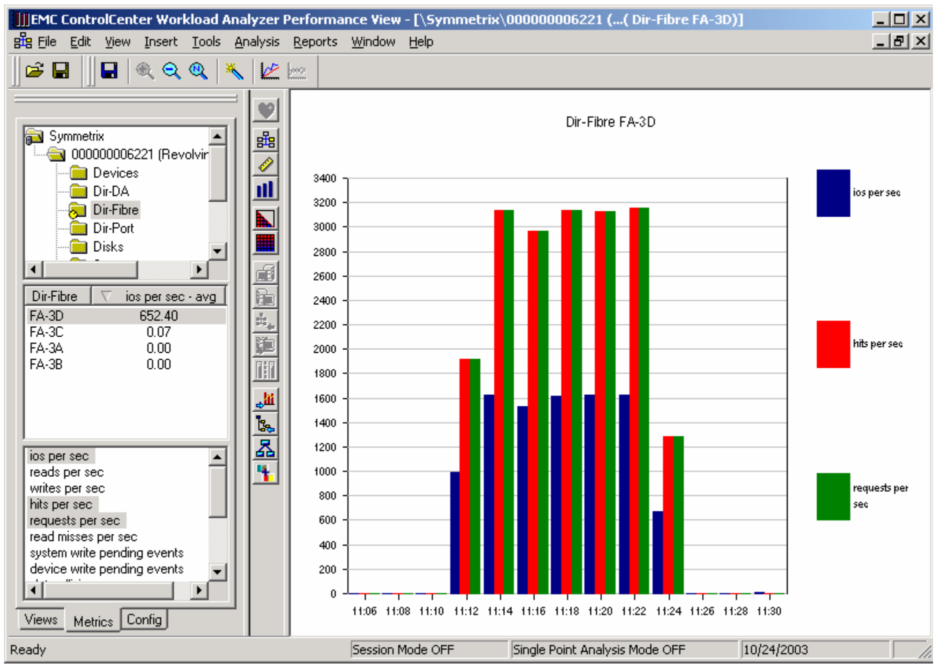
**Figure 3-10**

Notice that time spent at the host and the application thread reduced the maximum possible rate by a factor of 15 percent.

### 8 KB Random Writes, Unaligned

The amount of work performed by the host director CPU depends on the number of cache accesses needed to satisfy an I/O operation. When the I/O data buffer is not aligned on cache boundaries, additional work needs to be done in order to acquire additional cache slots and this translates into longer CPU time at the host director.

The same 8 KB random write test was run, but this time the addresses of the I/Os were forced to be such that the data would span multiple cache slots. Misalignment is confirmed by host directors with the requests versus I/Os metrics.



| Dir-Fibre | ios per sec - avg |
|---|---|
| FA-3D | 652.40 |
| FA-3C | 0.07 |
| FA-3A | 0.00 |
| FA-3B | 0.00 |

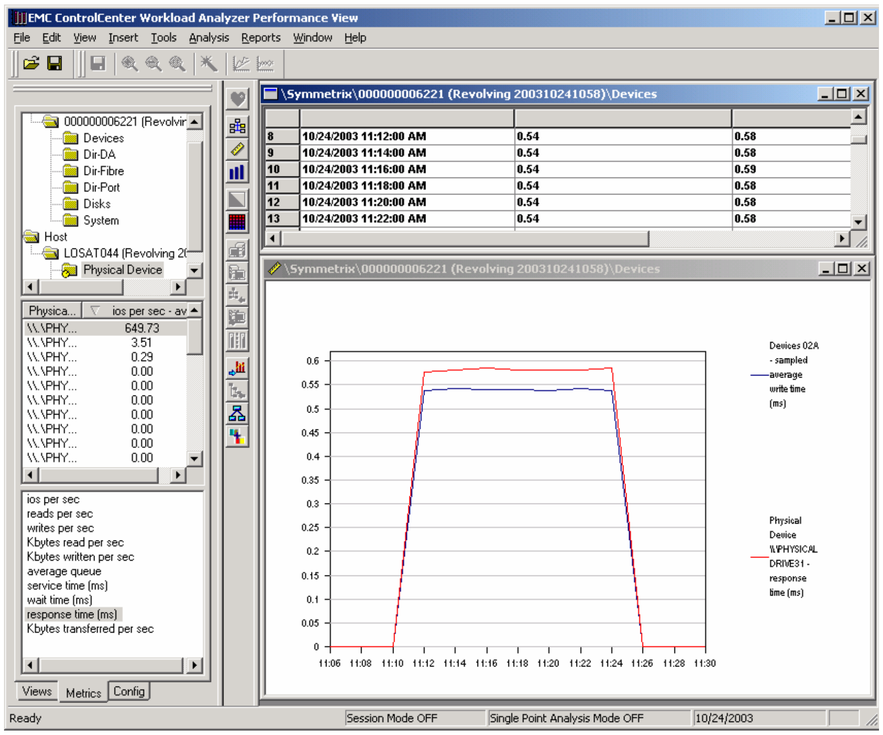
ios per sec
reads per sec
writes per sec
hits per sec
requests per sec
read misses per sec
system write pending events
device write pending events

Ready | Session Mode OFF | Single Point Analysis Mode OFF | 10/24/2003

**Figure 3-11**

Figure 3-11 shows a graph of host director activity. Three metrics are displayed. They are ios per sec, hits per sec, and requests per sec. Notice that the director reports exactly twice as many requests as I/Os. Also note that workload characteristics such as hits and misses (as well as reads and writes) as reported by the director, all refer to cache activity or *requests*. Therefore, as the graph shows, the number of hits equals the number of requests.

Since we see that the director is doing more work on behalf of each I/O, we anticipate two things: first, we expect the I/O rate to be lower,and second, we expect the response time to be higher.


EMC ControlCenter Workload Analyzer Performance View
File
Edit
View
Insert
Tools
Analysis
Reports
Window
Help
000000006221 (Revolvir
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 20
Physical Device
\Symmetrix\000000006221 (Revolving 200310241058)\Devices
\Symmetrix\000000006221 (Revolving 200310241058)\Devices
Devices 02A - sampled average write time (ms)
Physical Device \\.\PHYSICAL DRIVE31 - response time (ms)
Views
Metrics
Config
Ready
Session Mode OFF
Single Point Analysis Mode OFF
10/24/2003

| | | | |
|---|---|---|---|
| 8 | 10/24/2003 11:12:00 AM | 0.54 | 0.58 |
| 9 | 10/24/2003 11:14:00 AM | 0.54 | 0.58 |
| 10 | 10/24/2003 11:16:00 AM | 0.54 | 0.59 |
| 11 | 10/24/2003 11:18:00 AM | 0.54 | 0.58 |
| 12 | 10/24/2003 11:20:00 AM | 0.54 | 0.58 |
| 13 | 10/24/2003 11:22:00 AM | 0.54 | 0.58 |

| Physica... | ios per sec - av |
|---|---|
| \\.\PHY... | 649.73 |
| \\.\PHY... | 3.51 |
| \\.\PHY... | 0.29 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |

ios per sec
reads per sec
writes per sec
Kbytes read per sec
Kbytes written per sec
average queue
service time (ms)
wait time (ms)
response time (ms)
Kbytes transferred per sec

**Figure 3-12 Host Director Activity Graph**

Figure 3-12 shows the response times as reported by both the host and the Symmetrix array.

From the following table, we can see that the I/O rate was 1600 I/Os per second. This let us calculate the response time as seen by the application as 1000/1600 = .625 milliseconds.

We can now proceed to build the response time decomposition table below:

| | | Measured Response Time (in ms) | Layer Service Time (in ms) |
|---|---|---|---|
| Measured | Application time | 0.625 | .045. |
| | Host time | 0.58 | 0.04 |
| | Symmetrix | 0.54 | |
| Calculated | Bus (Figure 6) | 0.062 | 0.062 |
| | Host director | 0.478 | 0.478 |

Using these figures, we can now fill in the model to estimate the maximum I/O rate that we could achieve relative to the actually achieved rate.

| I/O Packet Size | Service Time on 1 Gb Fibre (in ms) | Service Time at the Host Director (in ms) | Total Service Time | Total Number of I/Os per Second | | % Time Bus Busy | % Time Director Busy | % Front-end Idle |
|---|---|---|---|---|---|---|---|---|
| 8192 | 0.06 | 0.478 | 0.54 | 1852 | Maximum | 11.48 | 88.52 | 0.00 |
| | | 0.478 | 0.54 | 1600 | Achieved | 9.92 | 76.48 | 13.60 |

A graph is created to show the calculated utilization levels showing the idle time at the front end.

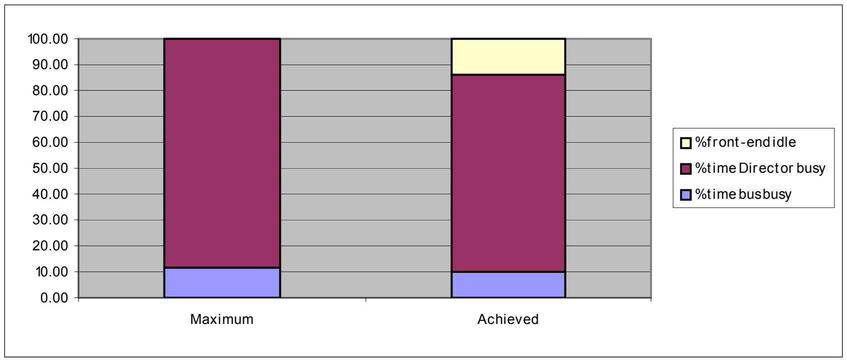


**Figure 3-13**

As can be seen from the table and the graph we have roughly 13 percent slack available in the front end. With multithreading, we should be able to achieve the maximum rate of around 1850 I/Os per second if the data is not aligned.

The next test uses the aligned case to confirm the predictions made by the model based on the values reported by WLA.

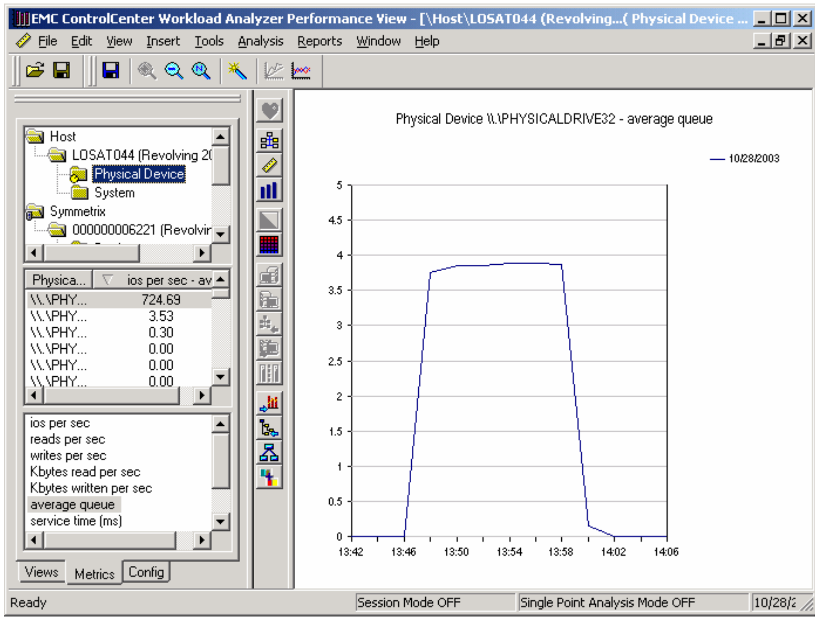
### 3.3.3.2 Multi-Threaded Synchronous Write-Hit, Aligned

We've seen that with a single-threaded synchronous I/O mode, we were not able to achieve maximum utilization of either bus or host director, nor was the sum of the two 100 percent. The work done at the host and the application created slack time at the front-end machine. The front-end machine as a whole was 16 percent idle (Figure 3-10 on page 3-12).

Using the total front-end I/O service time of .35 milliseconds and the available 16 percent idle time for the front end, we can estimate the potential limit of the front-end machine in a multithreaded environment.

| I/O Packet Size | Service Time on 1 Gb Fibre (in ms) | Service Time at the Host Director (in ms) | Total Service Time | Actual | % Time Bus Busy | % Time Director Busy | % Front-End Idle | Front-End Potential |
|---|---|---|---|---|---|---|---|---|
| 8192 | 0.062 | 0.288 | 0.35 | 2400 | 14.88 | 69.12 | 16.00 | 2857.14 |

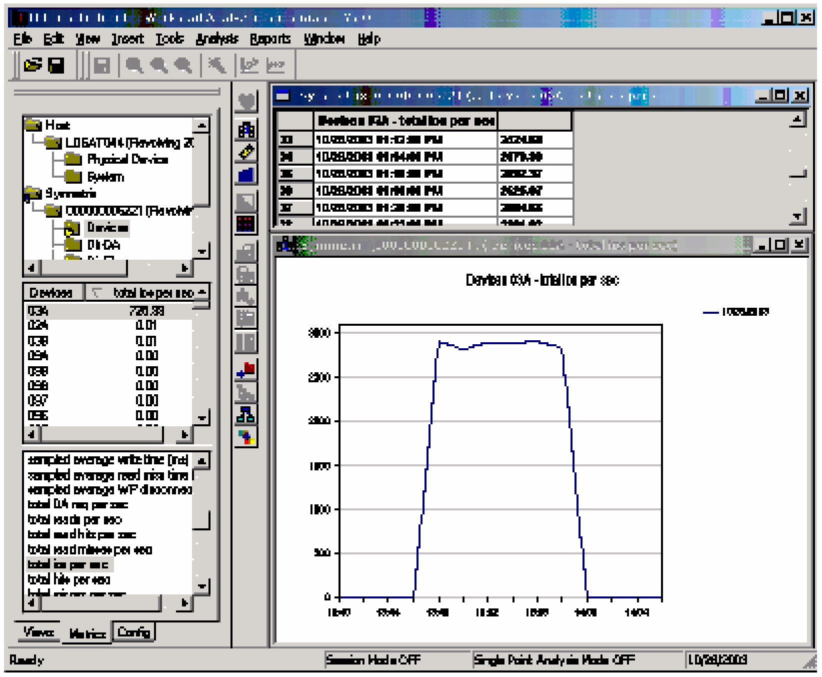
We ran our random write-hit test again with a thread count of four this time. The host statistics collected from our Windows-based host showed a queue depth of four to confirm our experiment.



| Physica... | ios per sec - av |
|---|---|
| \\.\PHY... | 724.69 |
| \\.\PHY... | 3.53 |
| \\.\PHY... | 0.30 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |

**Figure 3-14**

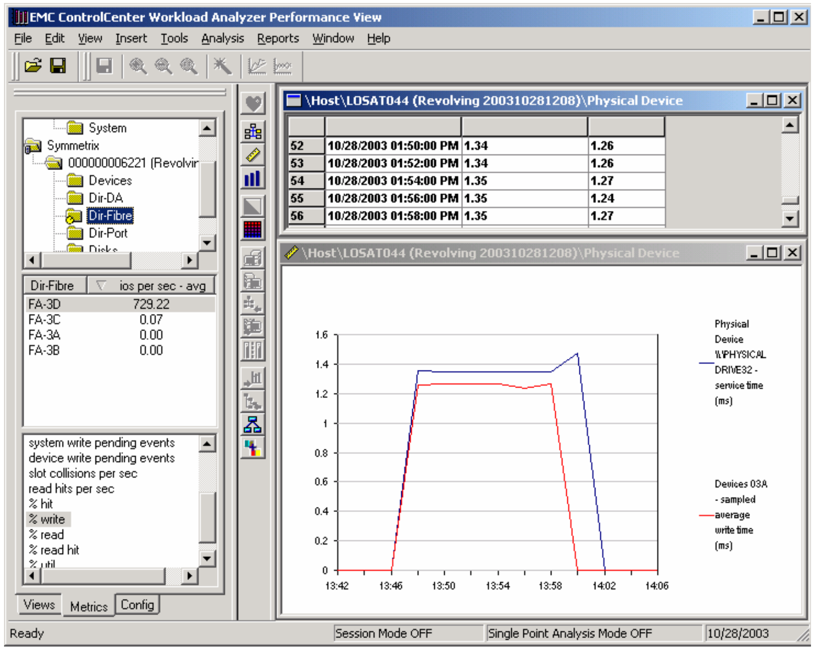
For a confirmation of the model, we need to look at the I/O rate (Figure 3-15).


File Edit View Insert Tools Analysis Reports Window Help
Host
LOSAT044 (Revolving 2(
Physical Device
System
Symmetrix
000000006221 (Revolvir
Devices
DI-DA
Devices total ios per sec
sampled average write time (ms)
sampled average read miss time
sampled average WP disconnect
total DA req per sec
total reads per sec
total read hits per sec
total read misses per sec
total ios per sec
total hits per sec
Views Metrics Config
Devices 03A - total ios per sec
Device 03A - total ios per sec
3000
2500
2000
1500
1000
500
0
Ready
Session Mode OFF
Single Point Analysis Mode OFF
10/28/2003

| Devices | total ios per sec |
|---|---|
| 03A | 720.33 |
| 024 | 0.01 |
| 038 | 0.01 |
| 054 | 0.00 |
| 058 | 0.00 |
| 056 | 0.00 |
| 037 | 0.00 |
| 055 | 0.00 |

**Figure 3-15**

Using the model, we estimated a value of 2857 I/Os per second. In actuality, the test peaked at 2892 I/Os per second. Response times, however, are now a function of the queue depth.


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
System
Symmetrix
000000006221 (Revolvir
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
system write pending events
device write pending events
slot collisions per sec
read hits per sec
% hit
% write
% read
% read hit
% util
Views
Metrics
Config
\Host\LOSAT044 (Revolving 200310281208)\Physical Device
\Host\LOSAT044 (Revolving 200310281208)\Physical Device
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
13:42
13:46
13:50
13:54
13:58
14:02
14:06
Physical Device \\.\PHYSICAL DRIVE32 - service time (ms)
Devices 03A - sampled average write time (ms)
Ready
Session Mode OFF
Single Point Analysis Mode OFF
10/28/2003

| Dir-Fibre | ios per sec - avg |
|---|---|
| FA-3D | 729.22 |
| FA-3C | 0.07 |
| FA-3A | 0.00 |
| FA-3B | 0.00 |

| | | | |
|---|---|---|---|
| 52 | 10/28/2003 01:50:00 PM | 1.34 | 1.26 |
| 53 | 10/28/2003 01:52:00 PM | 1.34 | 1.26 |
| 54 | 10/28/2003 01:54:00 PM | 1.35 | 1.27 |
| 55 | 10/28/2003 01:56:00 PM | 1.35 | 1.24 |
| 56 | 10/28/2003 01:58:00 PM | 1.35 | 1.27 |

**Figure 3-16**

We can see now that response times have increased substantially relative to the single threaded experiment. Symmetrix front-end response times now are clocked at a high of 1.27 milliseconds compared with previous values of .35 milliseconds. Using these two values we can approximate the queue depth:

1.27 / .35 = **3.62**

The following table shows the queue depth from the host perspective and the values are averaging around the 3.8 mark, which makes our estimates quite reasonable for a first approximation.

Another test was run with eight active threads to confirm that the front end had actually reached its capacity and thus validating the model. The key results of all three tests are as follows:

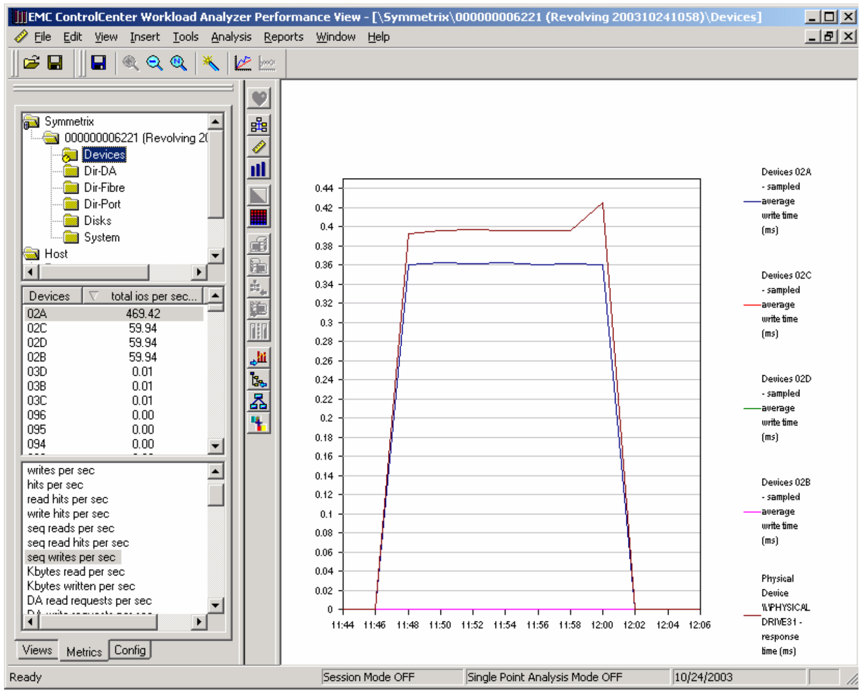
| # of Active Threads | Peak I/O Rate | Symmetrix Response Time (ms) | Estimated Queue Depth | Host Measured Queue Depth |
|---|---|---|---|---|
| 1 | | 0.35 | | |
| 4 | 2893 | 1.27 | 3.63 | 3.88 |
| 8 | 2892 | 2.72 | 7.77 | 7.86 |

## 3.3.4 Single-Threaded Sequential Writes

Write operations on the Symmetrix array are acknowledged back to the host as soon as they are stored in the cache. This action is independent of the destaging activity that ultimately stores the data on permanent disk storage. A write-hit occurs whenever a write operation immediately finds a cache slot and does not have to wait for space to be made available through a destaging operation.

Such operations apply to both random and sequential writes. Therefore, the overhead at the front end associated with processing a sequential-write operation is similar to that of a random write. There are no explicit counters in open systems environments that indicate sequential write activity. We will, however, show some indirect methods of determining when write activity is sequential.

Consistent with our techniques, we will focus on the response times measured by both the host and the Symmetrix array for our test. The test is an 8 KB sequential I/O stream which, in order to ensure unique addressing for the duration of the test, is run on a meta device with four members and a size of 17.6 GB.


EMC ControlCenter Workload Analyzer Performance View - [\Symmetrix\000000006221 (Revolving 200310241058)\Devices]
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221 (Revolving 20
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
Views
Metrics
Config
0.44
0.42
0.4
0.38
0.36
0.34
0.32
0.3
0.28
0.26
0.24
0.22
0.2
0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
11:44
11:46
11:48
11:50
11:52
11:54
11:56
11:58
12:00
12:02
12:04
12:06
Devices 02A - sampled average write time (ms)
Devices 02C - sampled average write time (ms)
Devices 02D - sampled average write time (ms)
Devices 02B - sampled average write time (ms)
Physical Device \\.\PHYSICALDRIVE31 - response time (ms)

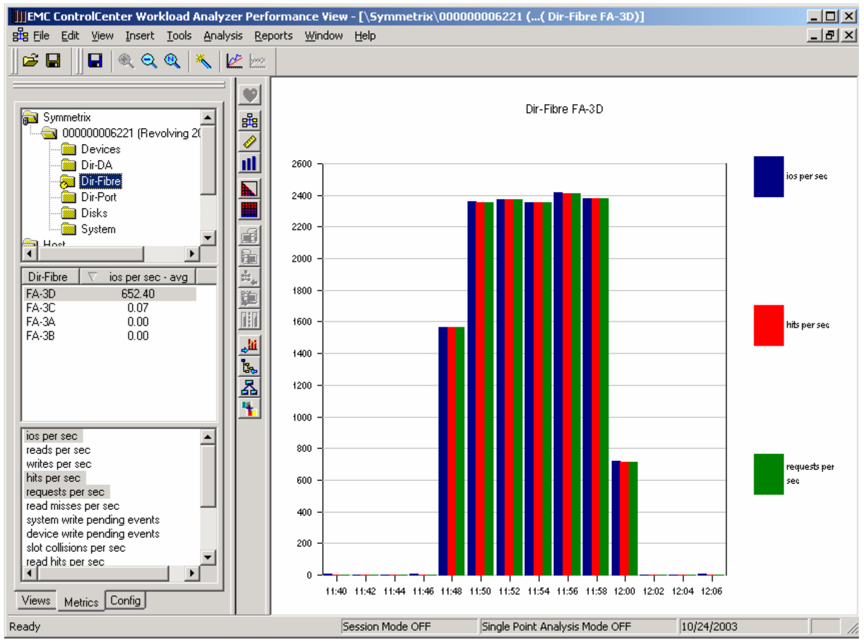
| Devices | total ios per sec... |
|---|---|
| 02A | 469.42 |
| 02C | 59.94 |
| 02D | 59.94 |
| 02B | 59.94 |
| 03D | 0.01 |
| 03B | 0.01 |
| 03C | 0.01 |
| 096 | 0.00 |
| 095 | 0.00 |
| 094 | 0.00 |

writes per sec
hits per sec
read hits per sec
write hits per sec
seq reads per sec
seq read hits per sec
seq writes per sec
Kbytes read per sec
Kbytes written per sec
DA read requests per sec

Ready | Session Mode OFF | Single Point Analysis Mode OFF | 10/24/2003

**Figure 3-17**

We can first verify that the writes are aligned and that we are not hampered by any write-pending conditions. This is done by ensuring that the I/O rate is equal to the request rate and the hit rate (Figure 3-19).



| Dir-Fibre | ios per sec - avg |
|---|---|
| FA-3D | 652.40 |
| FA-3C | 0.07 |
| FA-3A | 0.00 |
| FA-3B | 0.00 |

**Figure 3-18**

The achieved I/O rate for this test is around 2350 I/Os per second. This let us calculate the response time as seen by the application as 1000/2350 = .425 milliseconds.

We can now proceed to build the response time decomposition table (Table 3-9).

**Table 3-1**

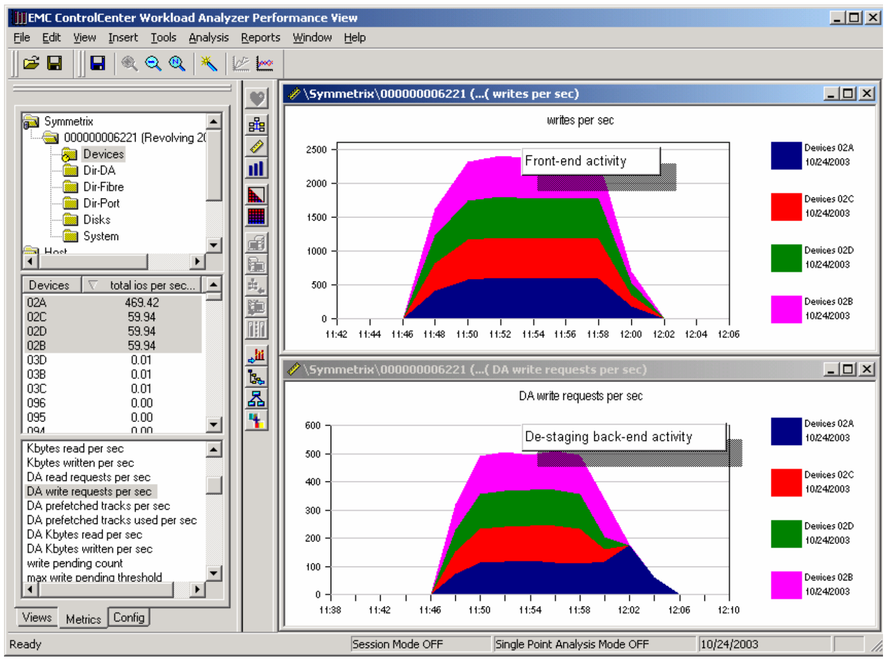
| | | | Measured Response Time (in ms) | Layer Service Time (in ms) |
|---|---|---|---|---|
| Measured | Application Time | | 0.425 | .36 |
| | Host Time | | 0.39 | 0.03 |
| | Symmetrix | | 0.36 | |
| Calculated | | Bus | 0.062 | 0.062 |
| | | Host director | 0.3 | 0.3 |

Table 3-10 shows the model for the maximum expected single-threaded sequential write rate.

**Table 3-2**

| I/O Packet Size | Service Time on 1 Gb Fibre (in ms) | Service Time at the Host Director (in ms) | Total Service Time | Total Number of I/Os per Second | | % Time Bus Busy | % Time Director Busy | % Front-End Idle |
|---|---|---|---|---|---|---|---|---|
| 8192 | 0.06 | 0.3 | 0.36 | 2762 | Maximum | 17.13 | 82.87 | 0.00 |
| | | 0.3 | 0.36 | 2350 | Achieved | 14.57 | 70.50 | 14.93 |

We ran the random write tests over a relatively small data address range. Therefore, we experienced many overwrites, and thus no destaging activity. However, with the sequential write over an address range of 17.6 GB, we began to see the destaging taking place. It is in this activity that we are able to deduce the sequential nature of the I/O.


EMC ControlCenter Workload Analyzer Performance View

Symmetrix
000000006221 (Revolving 2(
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host

| Devices | total ios per sec... |
|---|---|
| 02A | 469.42 |
| 02C | 59.94 |
| 02D | 59.94 |
| 02B | 59.94 |
| 03D | 0.01 |
| 03B | 0.01 |
| 03C | 0.01 |
| 096 | 0.00 |
| 095 | 0.00 |
| 094 | 0.00 |

Kbytes read per sec
Kbytes written per sec
DA read requests per sec
DA write requests per sec
DA prefetched tracks per sec
DA prefetched tracks used per sec
DA Kbytes read per sec
DA Kbytes written per sec
write pending count
max write pending threshold

Ready | Session Mode OFF | Single Point Analysis Mode OFF | 10/24/2003

**Figure 3-19**

The device metrics contain information about front-end activity as well as back-end activity. The DA write requests metric counts the number of times the back-end directors access the cache to extract the written data.

We saw that the front end was processing 2350 I/Os per second for the sequential writes. Yet at the back end, we see a rate of about 500 requests per second. We know that requests address a 32 KB cache slot. A sequential stream of 8 KB I/Os would take four I/Os to fill a cache slot. As a result, we would expect a rate of roughly 600 I/Os per second on the back end.

Destaging, however, is running at a slower pace since it is essentially running as fast as the disks can accept the data. We notice that while the front-end activity shows that the experiment ended at 12:02, the back end continued to write until 12:06. This additional time is used to make up for the slower speed.

The most important points in this test were:

- The response time for sequential write was approximately the same as for the random write of the same size.
- The back-end activity associated with destaging the written data did not impact the response times at all. This is not to say that in a full workload mix such activity would not impact any response times, but by itself it runs independently of the front-end activity for the same device.

## 3.3.5 Disclaimer

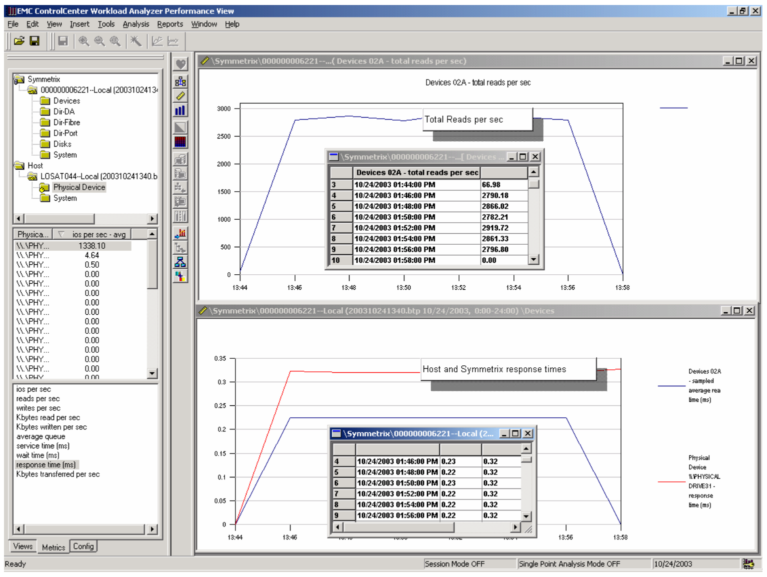
The values calculated in the experiments reflect the values of the specific workload generated during the tests. Host director CPU time per I/O type will vary depending on multiple variables. The following conditions have shown to generate different atomic unit costs (CPU time per I/O):

- Stand-alone devices vs. metavolumes, which would require the director to do more work in order to determine which meta member to use.
- The size of the devices, which would affect the write-pending limits for these devices and thus would generate write-miss conditions
- The overall I/O rate, which would impact the speed at which sequential writes enter the system and may or may not be impacted by write-pending limits

## 3.3.6 Random Read-Hits, Aligned

### 3.3.6.1 Single Threaded

Our random read-hit experiment is similar to the random-write one. The I/O size is 8 KB. It is a single-threaded job running on the host ensuring that no queues are built. As before, the goal is to analyze the reported response times and to use these values to characterize the work done by the front end.



EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221--Local (2003102413
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044--Local (200310241340.b
Physical Device
System
ios per sec
reads per sec
writes per sec
Kbytes read per sec
Kbytes written per sec
average queue
service time (ms)
wait time (ms)
response time (ms)
Kbytes transferred per sec
Views
Metrics
Config
\Symmetrix\000000006221--...( Devices 02A - total reads per sec)
Devices 02A - total reads per sec
Total Reads per sec
3000
2500
2000
1500
1000
500
0
13:44
13:46
13:48
13:50
13:52
13:54
13:56
13:58
\Symmetrix\000000006221--Local (200310241340.btp 10/24/2003, 0:00-24:00) \Devices
Host and Symmetrix response times
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
Devices 02A - sampled average read time (ms)
Physical Device \\.\PHYSICAL DRIVE31 - response time (ms)
Ready
Session Mode OFF
Single Point Analysis Mode OFF
10/24/2003

| Physica... | ios per sec - avg |
|---|---|
| \\.\PHY... | 1338.10 |
| \\.\PHY... | 4.64 |
| \\.\PHY... | 0.50 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |
| \\.\PHY... | 0.00 |

\Symmetrix\000000006221--...[ Devices ...

| | Devices 02A - total reads per sec | |
|---|---|---|
| 3 | 10/24/2003 01:44:00 PM | 66.98 |
| 4 | 10/24/2003 01:46:00 PM | 2790.18 |
| 5 | 10/24/2003 01:48:00 PM | 2866.02 |
| 6 | 10/24/2003 01:50:00 PM | 2782.21 |
| 7 | 10/24/2003 01:52:00 PM | 2919.72 |
| 8 | 10/24/2003 01:54:00 PM | 2861.33 |
| 9 | 10/24/2003 01:56:00 PM | 2796.80 |
| 10 | 10/24/2003 01:58:00 PM | 0.00 |

\Symmetrix\000000006221--Local (2...

| | | | |
|---|---|---|---|
| 4 | 10/24/2003 01:46:00 PM | 0.23 | 0.32 |
| 5 | 10/24/2003 01:48:00 PM | 0.22 | 0.32 |
| 6 | 10/24/2003 01:50:00 PM | 0.23 | 0.32 |
| 7 | 10/24/2003 01:52:00 PM | 0.22 | 0.32 |
| 8 | 10/24/2003 01:54:00 PM | 0.22 | 0.32 |
| 9 | 10/24/2003 01:56:00 PM | 0.22 | 0.32 |

**Figure 3-20**

We use the single threaded throughput to calculate the application response time:

The peak recorded value was 2919 I/Os per second. This translated to .342 milliseconds.

The host recorded .32 milliseconds and the Symmetrix .22 milliseconds, so our response- time table looks as follows:

**Table 3-3**

| | | | Measured Response Time (in ms) | Layer Service Time (in ms) |
|---|---|---|---|---|
| Measured | Application Time | | 0.342 | 0.022 |
| | Host Time | | 0.32 | 0.1 |
| | Symmetrix | | 0.22 | |
| Calculated | | Bus | 0.062 | 0.062 |
| | | Host Director | 0.16 | 0.16 |

Our quick estimator for the maximum throughput for this type of workload is:

**Table 3-4**

| I/O Packet Size | Service Time on 1 Gb Fibre (in ms) | Service Time at the Host Director (in ms) | Total Service Time | Total Number of I/Os per Second | | % Time Bus Busy | % Time Director Busy | % Front-end Idle |
|---|---|---|---|---|---|---|---|---|
| 8192 | 0.06 | 0.16 | 0.22 | 4505 | Maximum | 27.93 | 72.07 | 0.00 |
| | | 0.16 | 0.22 | 2919 | Achieved | 18.10 | 46.70 | 35.20 |

The calculated utilization charts for this estimation is:

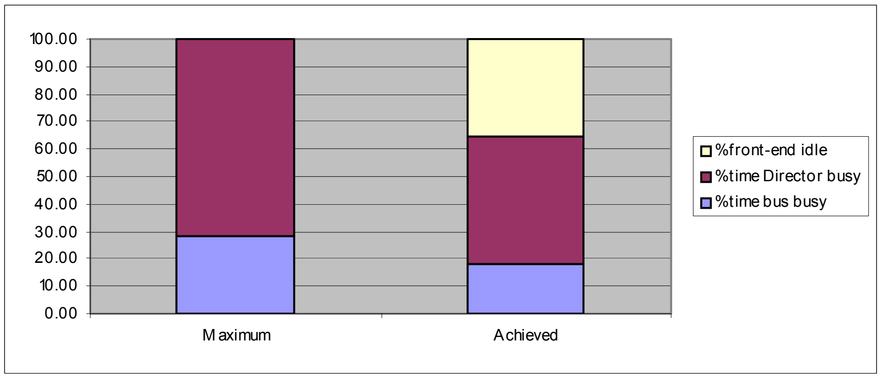
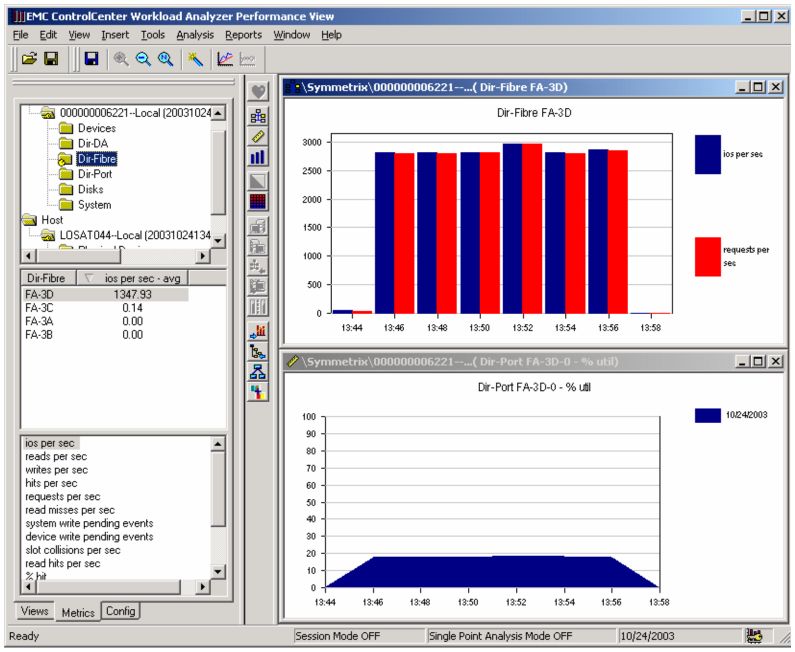


**Figure 3-21**

We check to ensure that the data is in fact aligned so that our calculated costs apply to a single cache request per I/O, and at the same time we can verify that our calculated port utilization is the same as that measured by WLA (Figure 3-22 on page 3-22).


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
000000006221--Local (20031024
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044--Local (20031024134
\Symmetrix\000000006221--...( Dir-Fibre FA-3D)
Dir-Fibre FA-3D
ios per sec
requests per sec
\Symmetrix\000000006221--...( Dir-Port FA-3D-0 - % util)
Dir-Port FA-3D-0 - % util
10/24/2003
Views
Metrics
Config
Ready
Session Mode OFF
Single Point Analysis Mode OFF
10/24/2003

| Dir-Fibre | ios per sec - avg |
|---|---|
| FA-3D | 1347.93 |
| FA-3C | 0.14 |
| FA-3A | 0.00 |
| FA-3B | 0.00 |

ios per sec
reads per sec
writes per sec
hits per sec
requests per sec
read misses per sec
system write pending events
device write pending events
slot collisions per sec
read hits per sec
% hit

Figure 3-22

## 3.3.7 Front-End Tests Summary

We've shown using a few experiments how the reported response time for devices can be used as a focal point around which other metrics are used to evaluate the well being of a system as well as characterize workloads for capacity planning. Not all combinations and permutations were shown since the intent of this document is to wet the readers' appetite to evaluate their own data.

## 3.4 The Back End

This section focuses on the activities that drive back-end activity. We will show how to evaluate the affect of the back end on the I/O response time if and when such effect occurs.

### 3.4.1 Command Tagged Queuing

The advent of command tag queuing in SCSI-2 revolutionized the protocol. It gave freedom to the SCSI target-LUN mechanism by permitting multiple I/O commands from the SCSI initiator to pass over the controller destined for the target. These SCSI commands dictate how to queue movement of associated data going to the LUN. There is an advantage of separating an I/O stream into smaller portions and managing them through a queue--the portions can be independently selected to move to the LUN based on a rule: LIFO, FIFO, or priority. The most economic rule involves efficient use of the disk by optimizing head movement as the I/O stream moves to the LUN.

The command tag queue feature adds a complexity to the disk I/O performance paradigm. It requires a new look at the performance calculations to get a firm understanding of the figures coming out of the various performance monitors.

#### 3.4.1.1 I/O Completion Time Metrics

In a multitiered environment where one component maintains a queue of requests for another component, the I/O-wait time includes in it the I/O active time at the subordinate component plus as any I/O-wait time experienced at that lower level.

Consider the I/O-wait time at the target, included in this figure is the io active time of the I/O at the disk plus the I/O-wait time for the disk level. This becomes an increasing factor as the number of multi-tiered environment where one resource maintains a queue of requests for another resource increases. This is the situation depicted by Figure 3-23. Multiple SCSI I/O commands are sent to the resource simultaneously. These can all be active at the same time. The traditional calculation of service time as I/O-wait time divided by the number of I/Os executed does not always show clearly in the observed results.

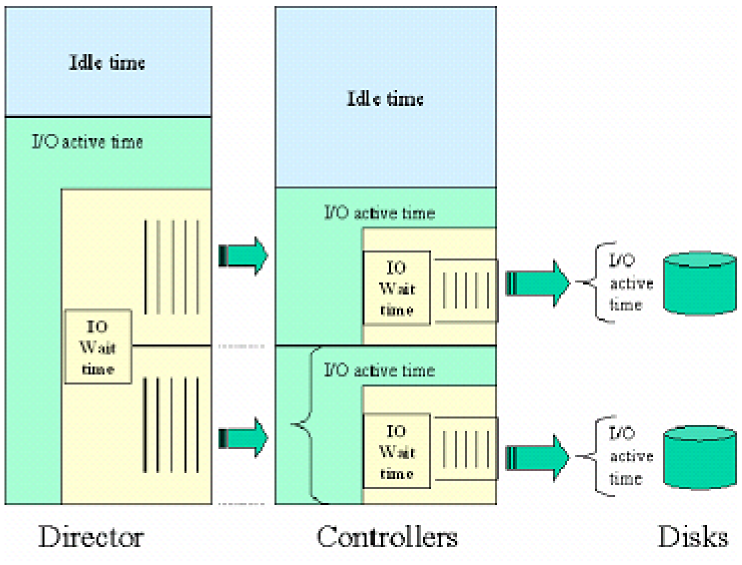


Figure 3-23

With the introduction of SCSI protocol came the ability to have multiple I/O commands sent to a component and active simultaneously. As a result, the traditional math that allowed us to divide the I/O-wait time by the number of I/Os executed in order to calculate the experienced service time does not always hold.

There are multiple interpretations of I/O completion times that are used interchangeably, on occasion, and that must be identified.

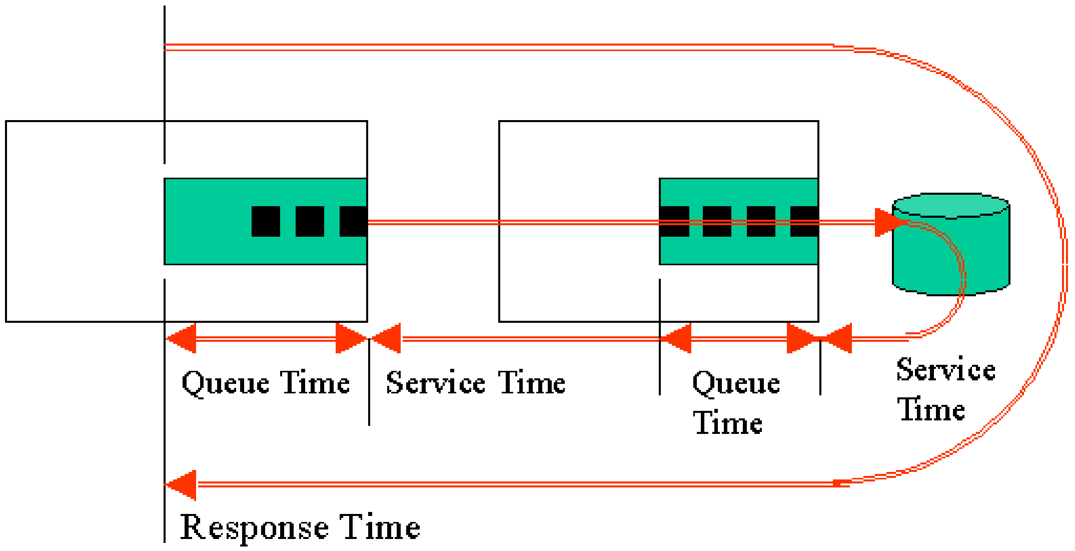


Figure 3-24

Response time is the metric most often associated with the end user who is the final recipient of the "answer". As Figure 3-24 shows, response time is a combination of service times and queue times where the service time in one place is actually a combination of queue times and service times at some other location. Depending on what is being measured and where, some of these measurements can be explicitly calculated while others may need to be estimated. In either case, an important aspect of systems that contain queues is the arrival pattern of requests. This is best described in terms of the relationship between the service time of an I/O and the inter-arrival time of the I/O requests. Three cases are discussed here:

- **Case 1** — The time between arrivals is, on average, greater than the service time. This means that the I/O completed before the next one arrived. The total I/O-wait time is the sum of the service times of the I/Os. In this case, the service time will never be equal to or greater than the total time interval.
- **Case 2** — The time between arrivals is, on average, equal to the service time. Because it is an average, queuing could occur. However in general, the average service time can be calculated as the I/O-wait time divided by the number of I/Os.
- **Case 3** — The time between arrivals is, on average, smaller than the service time. In this case, a queue is absolutely guaranteed. The I/O-wait time reflects the time starting with the first I/O until the completion of the last I/O in the queue. There are two subcases that need to be explored for this case.
- **Subcase A** — The I/Os are serial, meaning the first one must complete before the next is sent. In this case, the queued I/Os are maintained at the upstream SCSI resource and the calculations are identical to case 2. Service time contains the total amount of time needed to position the disk heads over the data. Rotational latency and individual seeks make up the bulk of the service time for each I/O.

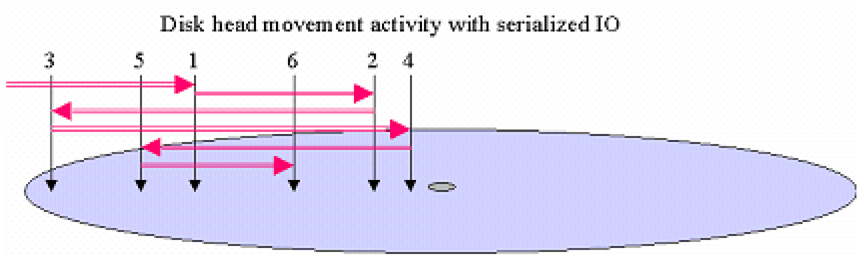


Figure 3-25

If six I/Os arrive at the director resource together, the inter-arrival time is 0. If the disk controller is servicing one I/O at a time, this table displays the measured and calculated values.

| I/O Ordered by Arrival | Disk I/O Service Time | Director Wait Time | Director Measured Response Time per I/O | Total I/O Busy = Elapsed Time |
|---|---|---|---|---|
| 1st | 10 | 0 | 10 | 10 |
| 2nd | 11 | 10 | 21 | 21 |
| 3rd | 9 | 21 | 30 | 30 |
| 4th | 12 | 30 | 42 | 42 |
| 5th | 10 | 42 | 52 | 52 |
| 6th | 8 | 52 | 60 | 60 |

| | |
|---|---|
| Cumulative measured response time | 215 |
| Average measured response time | 35.83 |
| Total I/O-wait time | 60 |
| Average calculated service time (I/O-wait / #I/Os) | 10 |
| Average calculated queue length | 3.58 |

- **Subcase B** — The feature known as command tagged queuing is applied, thereby allowing the device to receive multiple I/O requests at once or replenish ones already complete, and then reorder the requests to optimize the completion of the full I/O set. The optimization of the I/Os is done at the disk controller resource where the addresses of the I/Os determine their location on the physical disk. Re-ordering the I/Os based on physical addressing provides minimal (or even eliminates) seek and rotational latency times.

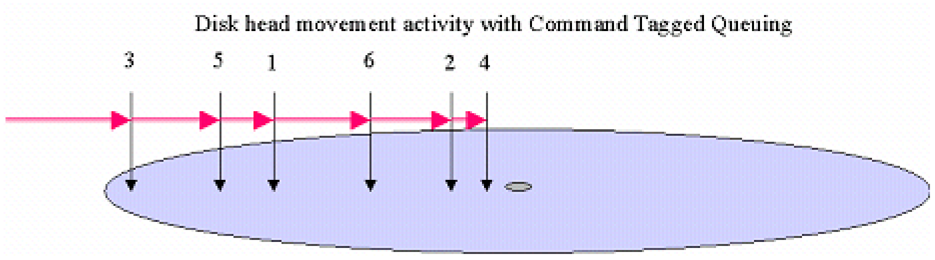


**Figure 3-26**

While this feature optimizes overall throughput in the system, it does present an accounting problem. It means that the total I/O-wait time and I/O rates are not sufficient metrics to determine the average service times and queue depth for the resource. Consider the situation that six I/Os entered the queue together and are reordered based on their addresses. Assume that the first I/O in the queue takes 10 ms and due to the realignment of the I/Os, subsequent I/Os each take less time to complete. Each must wait for all previous I/Os to complete. As a result, each I/Os service time includes the queue time it incurs at the disk controller.

If the total I/O-wait time for the five remaining I/Os, as measured outside the controller, was 30 milliseconds, then the standard calculation would estimate that the average service time per I/O to be 6 milliseconds. However, the component that sent the 6 I/Os to the controller would see 5 individual service times milliseconds (if it was measuring the service times), four of which would exceed 10 ms because they were in line behind the first I/O that took 10 ms.

**Table 3-5**

| Reordered I/O | Disk I/O Service Time | Disk Controller Wait Time | Externally Measured Service Time per I/O | Total I/O Busy = Elapsed Time |
|---|---|---|---|---|
| 1st | 10 | 0 | 10 | 10 |
| 2nd | 6 | 10 | 16 | 16 |
| 3rd | 4 | 16 | 20 | 20 |
| 4th | 5 | 20 | 25 | 25 |
| 5th | 5 | 25 | 30 | 30 |
| 6th | 3 | 30 | 33 | 33 |

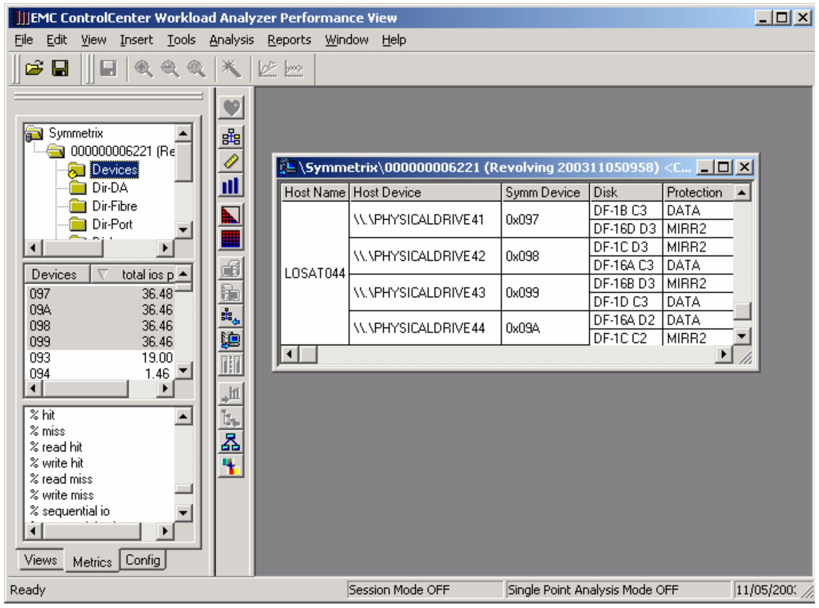
| | |
|---|---|
| Cumulative measured service time | 134 |
| Average measured service time | 22.33 |
| Total I/O-wait time | 33 |
| Average calculated I/O-wait time (service time) | 5.5 |
| Average calculated queue length | 4.06 |

The calculated queue length is the ratio of the cumulative measured service time and the total I/O-wait time or the ratio of the average measured service time and the average calculated io-wait time per I/O.

## 3.4.2 Destaging: Multithreaded Random Writes with Write-Misses

Four threads are started issuing random 8 KB writes to four devices. The purpose of this test is to generate a sufficient number of write-pending slots in cache for the devices so as to show the effect of write misses on the response time and to view the process of destaging by the back-end directors to the disks.

Figure 3-27 shows host/Symmetrix device configuration.


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221 (Re
Devices
Dir-DA
Dir-Fibre
Dir-Port
Views Metrics Config
\Symmetrix\000000006221 (Revolving 200311050958) <C...

| Devices | total ios p |
|---|---|
| 097 | 36.48 |
| 09A | 36.46 |
| 098 | 36.46 |
| 099 | 36.46 |
| 093 | 19.00 |
| 094 | 1.46 |

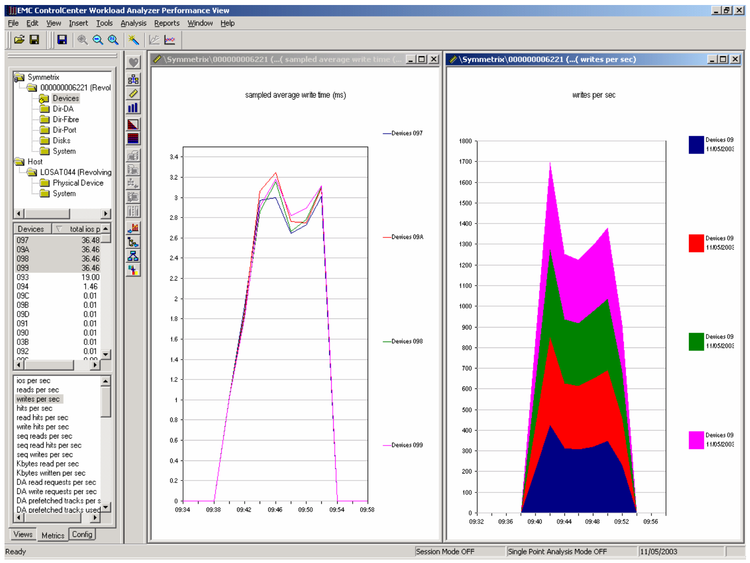
% hit
% miss
% read hit
% write hit
% read miss
% write miss
% sequential io

| Host Name | Host Device | Symm Device | Disk | Protection |
|---|---|---|---|---|
| LOSAT044 | \\.\PHYSICALDRIVE41 | 0x097 | DF-1B C3 | DATA |
| | | | DF-16D D3 | MIRR2 |
| | \\.\PHYSICALDRIVE42 | 0x098 | DF-1C D3 | MIRR2 |
| | | | DF-16A C3 | DATA |
| | \\.\PHYSICALDRIVE43 | 0x099 | DF-16B D3 | MIRR2 |
| | | | DF-1D C3 | DATA |
| | \\.\PHYSICALDRIVE44 | 0x09A | DF-16A D2 | DATA |
| | | | DF-1C C2 | MIRR2 |

Ready | Session Mode OFF | Single Point Analysis Mode OFF | 11/05/200

**Figure 3-27**

As with previous tests, we will start by viewing the I/O response times and the I/O rates.



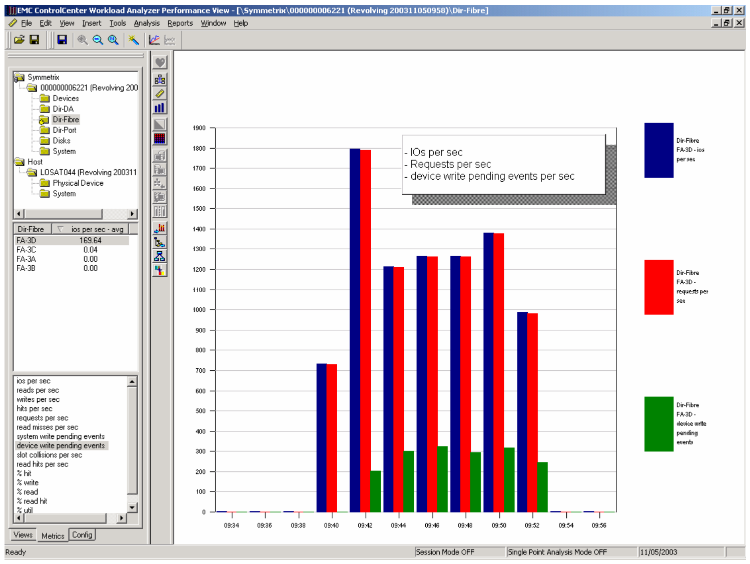
| Devices | total ios p |
|---|---|
| 097 | 36.48 |
| 09A | 36.46 |
| 098 | 36.46 |
| 099 | 36.46 |
| 093 | 19.00 |
| 094 | 1.46 |
| 09C | 0.01 |
| 09B | 0.01 |
| 09D | 0.01 |
| 091 | 0.01 |
| 090 | 0.01 |
| 03B | 0.01 |
| 092 | 0.01 |

Figure 3-28

Observe that response times have now increased to multiple milliseconds (3+) as opposed to previously observed submillisecond values. Multiple factors are at play here, specifically the write-pending limits and queue depths.

### 3.4.2.1 Device Write Pending

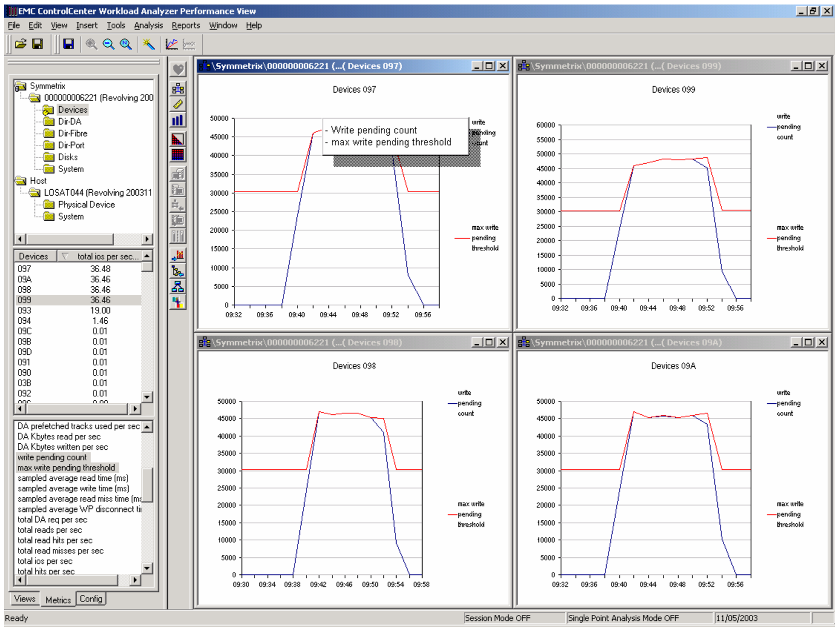
At the front end, we first confirm that the data is aligned as well as whether we are seeing write-pending events.


EMC ControlCenter Workload Analyzer Performance View - [\Symmetrix\000000006221 (Revolving 200311050958)\Dir-Fibre]
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221 (Revolving 200
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 200311
Physical Device
System
ios per sec
reads per sec
writes per sec
hits per sec
requests per sec
read misses per sec
system write pending events
device write pending events
slot collisions per sec
read hits per sec
% hit
% write
% read
% read hit
% util
Views
Metrics
Config
- IOs per sec
- Requests per sec
- device write pending events per sec
Dir-Fibre FA-3D - ios per sec
Dir-Fibre FA-3D - requests per sec
Dir-Fibre FA-3D - device write pending events
Ready
Session Mode OFF
Single Point Analysis Mode OFF
11/05/2003

| Dir-Fibre | ios per sec - avg |
|---|---|
| FA-3D | 169.64 |
| FA-3C | 0.04 |
| FA-3A | 0.00 |
| FA-3B | 0.00 |

Figure 3-29

We observe in Figure 3-30 on page 3-29 that the write-pending limits for the device were reached and that write misses began to occur. Figure 3-31 on page 3-29 shows how the percent write hit begins to degrade (graph on the right) the minute that the write-pending count reaches the write-pending limits of the device (graph on the left). Write misses begin to occur when the write-pending threshold is three times the base. As Figure 3-30 shows, that upper limit was not reached by any of the devices on a sustained basis, and therefore the percent write-miss rates are only around 25 percent.


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
\Symmetrix\000000006221 (...( Devices 097)
Devices 097
- Write pending count
- max write pending threshold
\Symmetrix\000000006221 (...( Devices 099)
Devices 099
\Symmetrix\000000006221 (...( Devices 098)
Devices 098
\Symmetrix\000000006221 (...( Devices 09A)
Devices 09A
write pending count
max write pending threshold
Views
Metrics
Config
Session Mode OFF
Single Point Analysis Mode OFF
11/05/2003

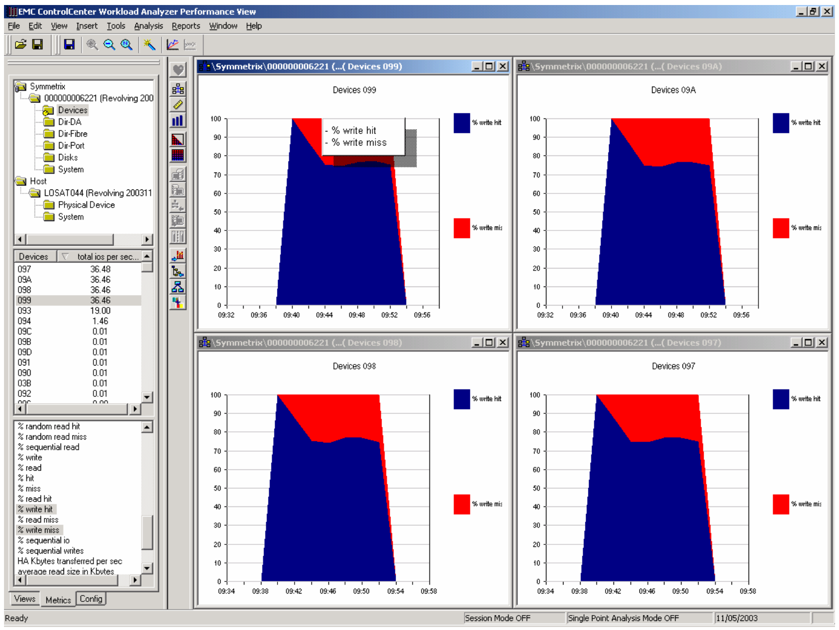
Symmetrix
000000006221 (Revolving 200
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 200311
Physical Device
System

| Devices | total ios per sec... |
|---|---|
| 097 | 36.48 |
| 09A | 36.46 |
| 098 | 36.46 |
| 099 | 36.46 |
| 093 | 19.00 |
| 094 | 1.46 |
| 09C | 0.01 |
| 09B | 0.01 |
| 09D | 0.01 |
| 091 | 0.01 |
| 090 | 0.01 |
| 03B | 0.01 |
| 092 | 0.01 |

DA prefetched tracks used per sec
DA Kbytes read per sec
DA Kbytes written per sec
write pending count
max write pending threshold
sampled average read time (ms)
sampled average write time (ms)
sampled average read miss time (ms
sampled average WP disconnect ti
total DA req per sec
total reads per sec
total read hits per sec
total read misses per sec
total ios per sec
total hits per sec

Ready

**Figure 3-30**


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
\Symmetrix\000000006221 (...( Devices 099)
Devices 099
- % write hit
- % write miss
\Symmetrix\000000006221 (...( Devices 09A)
Devices 09A
\Symmetrix\000000006221 (...( Devices 098)
Devices 098
\Symmetrix\000000006221 (...( Devices 097)
Devices 097
% write hit
% write mis
Views
Metrics
Config
Session Mode OFF
Single Point Analysis Mode OFF
11/05/2003

Symmetrix
000000006221 (Revolving 200
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 200311
Physical Device
System

| Devices | total ios per sec... |
|---|---|
| 097 | 36.48 |
| 09A | 36.46 |
| 098 | 36.46 |
| 099 | 36.46 |
| 093 | 19.00 |
| 094 | 1.46 |
| 09C | 0.01 |
| 09B | 0.01 |
| 09D | 0.01 |
| 091 | 0.01 |
| 090 | 0.01 |
| 03B | 0.01 |
| 092 | 0.01 |

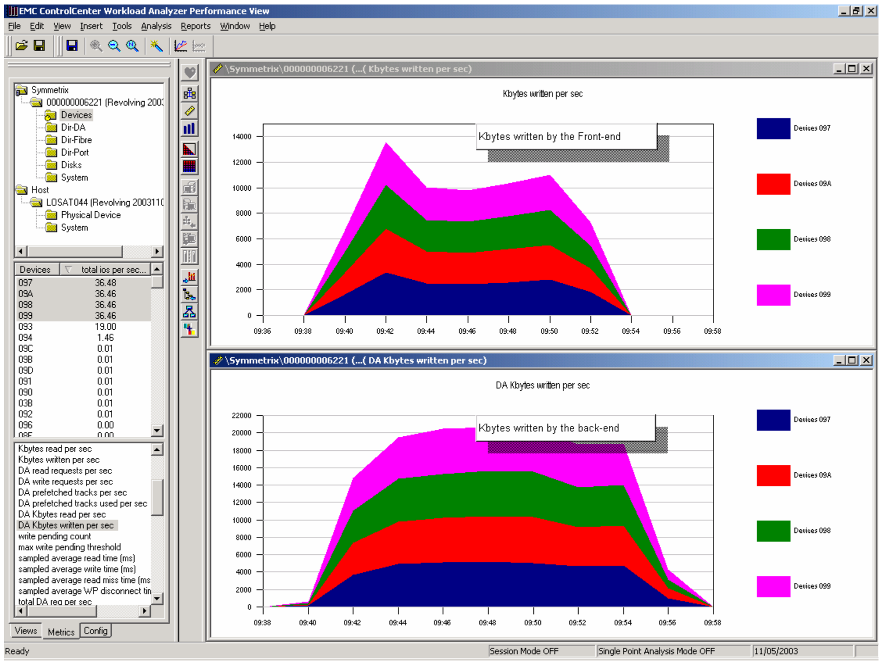
% random read hit
% random read miss
% sequential read
% write
% read
% hit
% miss
% read hit
% write hit
% read miss
% write miss
% sequential io
% sequential writes
HA Kbytes transferred per sec
average read size in Kbytes

Ready

**Figure 3-31**

### 3.4.2.2 Back-End Activity

As data is written, at some point the back end begins to destage the written data to the disks. Device statistics are collected for both the front-end and back-end activity. One set of metrics compares the amount of data written by the front end and the back end.


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221 (Revolving 200
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 2003110
Physical Device
System
Symmetrix\000000006221 (...( Kbytes written per sec)
Kbytes written per sec
Kbytes written by the Front-end
Devices 097
Devices 09A
Devices 098
Devices 099
Symmetrix\000000006221 (...( DA Kbytes written per sec)
DA Kbytes written per sec
Kbytes written by the back-end
Views
Metrics
Config
Ready
Session Mode OFF
Single Point Analysis Mode OFF
11/05/2003

| Devices | total ios per sec... |
|---|---|
| 097 | 36.48 |
| 09A | 36.46 |
| 098 | 36.46 |
| 099 | 36.46 |
| 093 | 19.00 |
| 094 | 1.46 |
| 09C | 0.01 |
| 09B | 0.01 |
| 09D | 0.01 |
| 091 | 0.01 |
| 090 | 0.01 |
| 03B | 0.01 |
| 092 | 0.01 |
| 096 | 0.00 |

Kbytes read per sec
Kbytes written per sec
DA read requests per sec
DA write requests per sec
DA prefetched tracks per sec
DA prefetched tracks used per sec
DA Kbytes read per sec
DA Kbytes written per sec
write pending count
max write pending threshold
sampled average read time (ms)
sampled average write time (ms)
sampled average read miss time (ms
sampled average WP disconnect tin
total DA req per sec

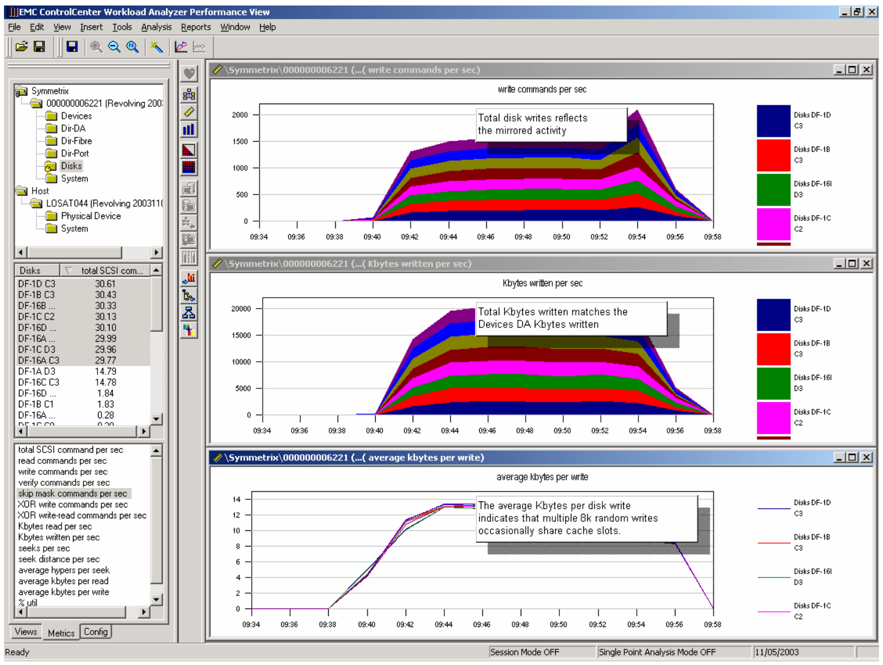
**Figure 3-32**

We observe the following:

- The total kilobytes destaged by the front end is approximately double the data received by the front end. This reflects the fact that the disk protection used for these devices is RAID 1 or mirrored.
- We can see that the destaging lags somewhat behind the front-end activity reflecting the fact that the disks are generally slower than the busses and cache.

For disk activity, we see:

- Eight disks active for four devices confirming the RAID 1 environment
- The payload (kilobytes per second) matches the devices back-end payload
- The size of each disk I/O indicates that on many occasions a cache slot is used by multiple I/Os


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221 (Revolving 200
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 200311
Physical Device
System
\Symmetrix\000000006221 (...( write commands per sec)
write commands per sec
\Symmetrix\000000006221 (...( Kbytes written per sec)
Kbytes written per sec
\Symmetrix\000000006221 (...( average kbytes per write)
average kbytes per write
Disks DF-1D C3
Disks DF-1B C3
Disks DF-16B D3
Disks DF-1C C2
Views
Metrics
Config
Ready
Session Mode OFF
Single Point Analysis Mode OFF
11/05/2003

Total disk writes reflects the mirrored activity

Total Kbytes written matches the Devices DA Kbytes written

The average Kbytes per disk write indicates that multiple 8k random writes occasionally share cache slots.

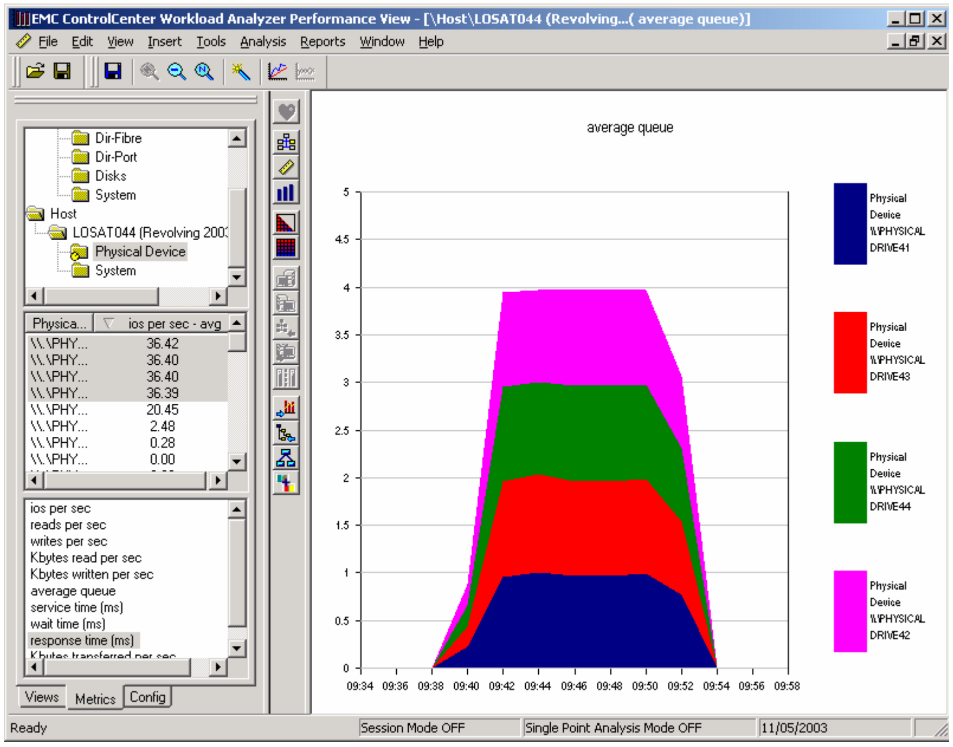
| Disks | total SCSI com... |
|---|---|
| DF-1D C3 | 30.61 |
| DF-1B C3 | 30.43 |
| DF-16B ... | 30.33 |
| DF-1C C2 | 30.13 |
| DF-16D ... | 30.10 |
| DF-16A ... | 29.99 |
| DF-1C D3 | 29.96 |
| DF-16A C3 | 29.77 |
| DF-1A D3 | 14.79 |
| DF-16C C3 | 14.78 |
| DF-16D ... | 1.84 |
| DF-1B C1 | 1.83 |
| DF-16A ... | 0.28 |

total SCSI command per sec
read commands per sec
write commands per sec
verify commands per sec
skip mask commands per sec
XOR write commands per sec
XOR write-read commands per sec
Kbytes read per sec
Kbytes written per sec
seeks per sec
seek distance per sec
average hypers per seek
average kbytes per read
average kbytes per write
% util

**Figure 3-33**

### 3.4.2.3 Response Time Decomposition

In our previous multithreaded write-hit test, we calculated that the host director's CPU time per I/O was .3 milliseconds. We would now like to decompose our WLA data in order to calculate the service time for a write miss. First, we need to recognize the queue depth due to the multithreading activity.


EMC ControlCenter Workload Analyzer Performance View - [\Host\LOSAT044 (Revolving...( average queue)]
File Edit View Insert Tools Analysis Reports Window Help
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 200
Physical Device
System
Views Metrics Config
average queue
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
09:34 09:36 09:38 09:40 09:42 09:44 09:46 09:48 09:50 09:52 09:54 09:56 09:58
Physical Device \\.\PHYSICAL DRIVE41
Physical Device \\.\PHYSICAL DRIVE43
Physical Device \\.\PHYSICAL DRIVE44
Physical Device \\.\PHYSICAL DRIVE42

| Physica... | ios per sec - avg |
|---|---|
| \\.\PHY... | 36.42 |
| \\.\PHY... | 36.40 |
| \\.\PHY... | 36.40 |
| \\.\PHY... | 36.39 |
| \\.\PHY... | 20.45 |
| \\.\PHY... | 2.48 |
| \\.\PHY... | 0.28 |
| \\.\PHY... | 0.00 |

ios per sec
reads per sec
writes per sec
Kbytes read per sec
Kbytes written per sec
average queue
service time (ms)
wait time (ms)
response time (ms)
Kbytes transferred per sec

Ready | Session Mode OFF | Single Point Analysis Mode OFF | 11/05/2003

**Figure 3-34**

Figure 3-34 shows the queue depth from the host perspective. The way that the test is conducted, four threads are started and each operates in a synchronous fashion such that as soon as one I/O ends another is sent. Therefore, the 3$^{+}$ milliseconds response times shown in Figure 3-33 reflect the wait in the queue as well as the service time.

Each thread shows a queue depth of 1 since the thread is synchronous. However, since all four threads target a single director CPU, the internal front-end queue depth is actually the sum of the threads.

We know that response time equals service time when there is no queue so the average service time is calculated as:

Response time / Queue depth = 3.2 / 4 = .8 milliseconds per I/O.

We would like to use the hit rate and the known host director CPU time to estimate the remaining back-end service time. The following formula is a start:

(% hit * hit service time) + ( (1-%hit)* miss service time) = average service time

From previous analysis and WLA we have the following:

% hit = 75%

% miss = 25%

hit service time = .0003

miss service time = ???

average service time = .0008

We will solve for the **miss service time**

(.75*.0003) + (.25*x) = .0008

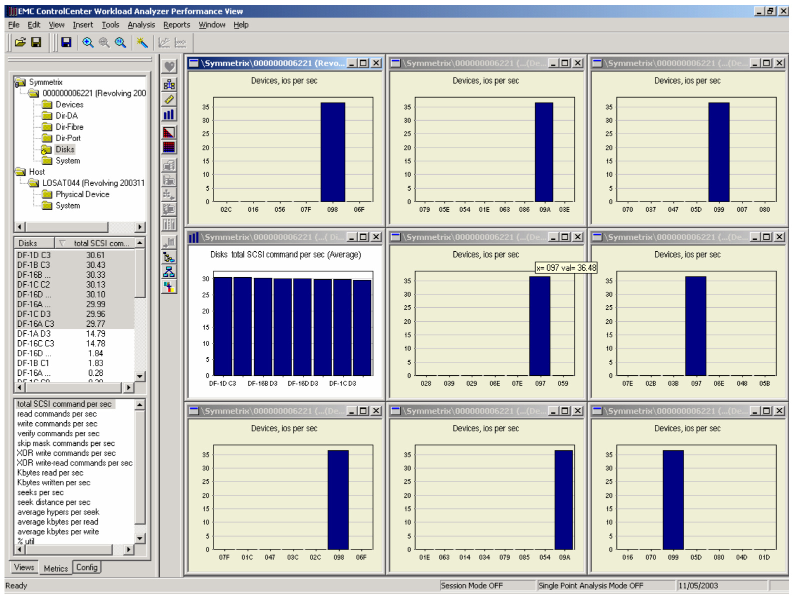
0.000225 + .25x = .0008

.25x = .0008-0.000225= 0.000575

x = **0.0023 = 2.3 milliseconds**

Removing the front-end cache service time of .3 milliseconds, we have 2 milliseconds of back-end time spent at the director and disk.

How do we explain such good disk response time?


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221 (Revolving 200
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 200311
Physical Device
System
Devices, ios per sec
Disks total SCSI command per sec (Average)
x= 097 val= 36.48
Views
Metrics
Config
Ready
Session Mode OFF
Single Point Analysis Mode OFF
11/05/2003

| Disks | total SCSI com... |
|---|---|
| DF-1D C3 | 30.61 |
| DF-1B C3 | 30.43 |
| DF-16B ... | 30.33 |
| DF-1C C2 | 30.13 |
| DF-16D ... | 30.10 |
| DF-16A ... | 29.99 |
| DF-1C D3 | 29.96 |
| DF-16A C3 | 29.77 |
| DF-1A D3 | 14.79 |
| DF-16C C3 | 14.78 |
| DF-16D ... | 1.84 |
| DF-1B C1 | 1.83 |
| DF-16A ... | 0.28 |

total SCSI command per sec
read commands per sec
write commands per sec
verify commands per sec
skip mask commands per sec
XOR write commands per sec
XOR write-read commands per sec
Kbytes read per sec
Kbytes written per sec
seeks per sec
seek distance per sec
average hypers per seek
average kbytes per read
average kbytes per write
% util

**Figure 3-35**

Using Performance View drill-down feature, we can drill down to each of the disks via the histogram view (white background) to see the devices at each disk as well as the device that receives the activity. We can see that each disk has multiple devices on it, but only one device is active. Therefore, the head movement on that disk is quite small, significantly speeding up disk I/Os.

## 3.4.3 Random Reads

The interesting aspect of random reads is observing and calculating the positive effects of command tagged queuing. As described in *3.4.1 Command Tagged Queuing* on page 3-23, this technique sends multiple I/O commands to the disk where the disk controller re-orders the I/Os based on physical location in order to optimize access.

We will use the same 8 KB test we've been using all along. We will run two tests:

- Single threaded

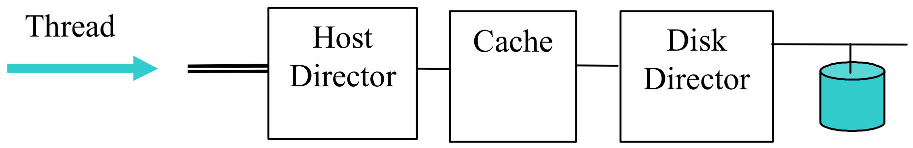


Figure 3-36

- Multithreaded to the same device

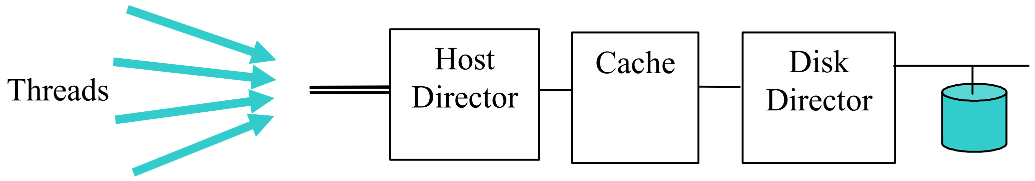
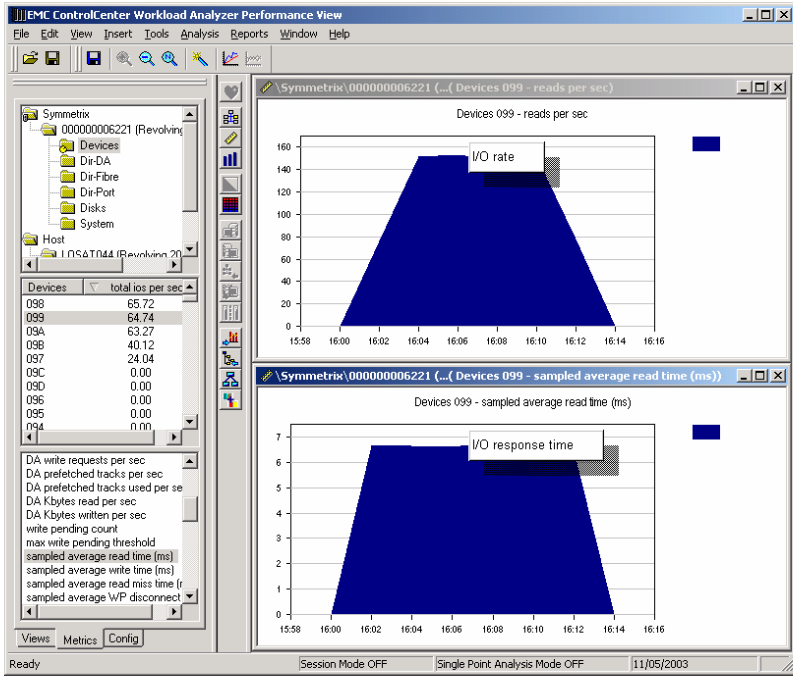


Figure 3-37

### 3.4.3.1 Single Threaded 8k Random Reads

We begin with the I/O rate and the response time.


EMC ControlCenter Workload Analyzer Performance View
File Edit View Insert Tools Analysis Reports Window Help
Symmetrix
000000006221 (Revolving
Devices
Dir-DA
Dir-Fibre
Dir-Port
Disks
System
Host
LOSAT044 (Revolving 20
Views Metrics Config
\Symmetrix\000000006221 (...( Devices 099 - reads per sec)
Devices 099 - reads per sec
I/O rate
\Symmetrix\000000006221 (...( Devices 099 - sampled average read time (ms))
Devices 099 - sampled average read time (ms)
I/O response time
Ready
Session Mode OFF
Single Point Analysis Mode OFF
11/05/2003

| Devices | total ios per sec |
|---|---|
| 098 | 65.72 |
| 099 | 64.74 |
| 09A | 63.27 |
| 09B | 40.12 |
| 097 | 24.04 |
| 09C | 0.00 |
| 09D | 0.00 |
| 096 | 0.00 |
| 095 | 0.00 |
| 094 | 0.00 |

DA write requests per sec
DA prefetched tracks per sec
DA prefetched tracks used per se
DA Kbytes read per sec
DA Kbytes written per sec
write pending count
max write pending threshold
sampled average read time (ms)
sampled average write time (ms)
sampled average read miss time (r
sampled average WP disconnect

Figure 3-38

Our first observation is that we are no longer working with thousands of I/Os per second but rather in hundreds. This is our first indication that the workload is throttled by the disk. The graph showing the hit and miss percentages is a clear indication.

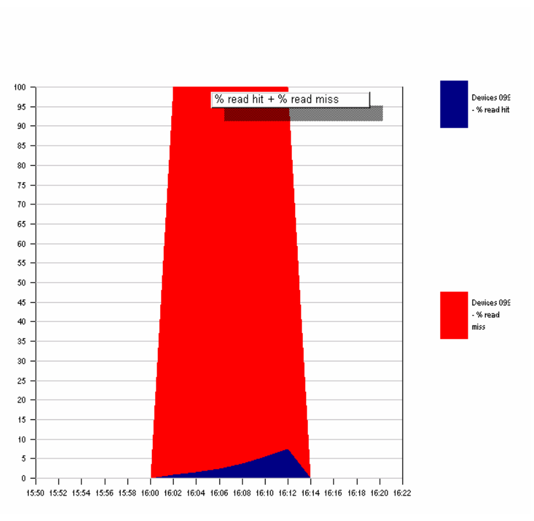
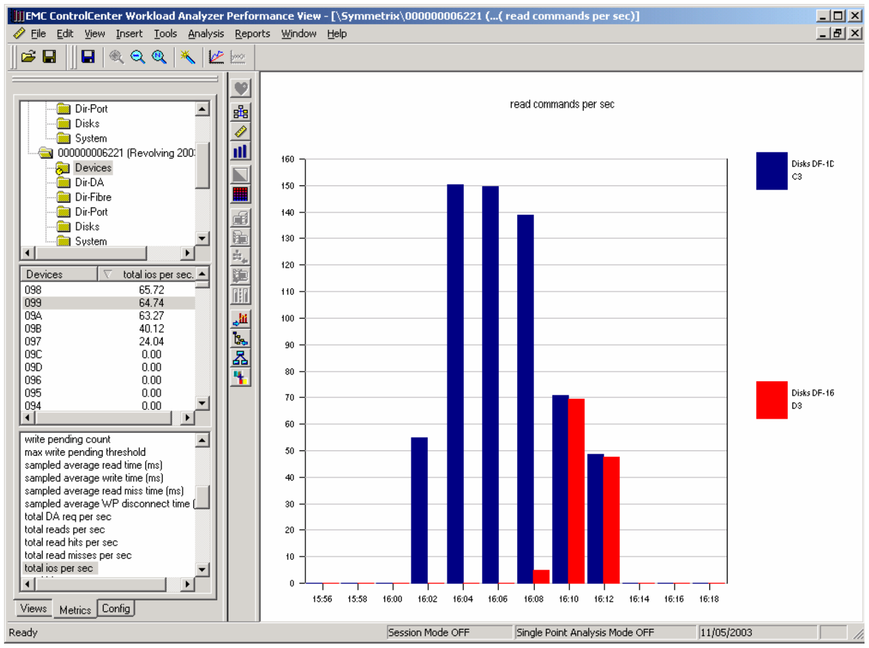


**Figure 3-39**

We use the configuration section to identify the disks that are associated with device 099 then, we check the disk I/O activity.


EMC ControlCenter Workload Analyzer Performance View - [\Symmetrix\000000006221 (...( read commands per sec)]

| Devices | total ios per sec. |
|---|---|
| 098 | 65.72 |
| 099 | 64.74 |
| 09A | 63.27 |
| 09B | 40.12 |
| 097 | 24.04 |
| 09C | 0.00 |
| 09D | 0.00 |
| 096 | 0.00 |
| 095 | 0.00 |
| 094 | 0.00 |

write pending count
max write pending threshold
sampled average read time (ms)
sampled average write time (ms)
sampled average read miss time (ms)
sampled average WP disconnect time (
total DA req per sec
total reads per sec
total read hits per sec
total read misses per sec
total ios per sec

Ready | Session Mode OFF | Single Point Analysis Mode OFF | 11/05/2003

**Figure 3-40**

It looks like both the data and mirror disks are active indicating that the Symmetrix Dynamic Mirror Service Policy (DMSP) is turned on. This internal algorithm checks the load on the disks and decides which disk is the most appropriate one to serve the reads to a particular device for a given time window.

The system was idle prior to the test, so the default DMSP algorithm was to alternate between the data disk and the mirror disk every 5 minutes. However, after the first 5 minutes of the test the back-end director, using the DMSP algorithm, determined that the alternating method was the best policy for this device, so we can see that during the later part of the test, both drives are active serving the requests.

This has an impact when many devices are active. In this test's case, there should be no impact since at any point in time there is only one I/O active. In the following multithreaded test, we will see how this algorithm improves throughput.

We would like to use the hit rate and the known host director CPU time to estimate the remaining back-end service time. The following formula is a start:

(% hit * hit service time) + ( (1-%hit)* miss service time) = average service time

From WLA, we have the following:

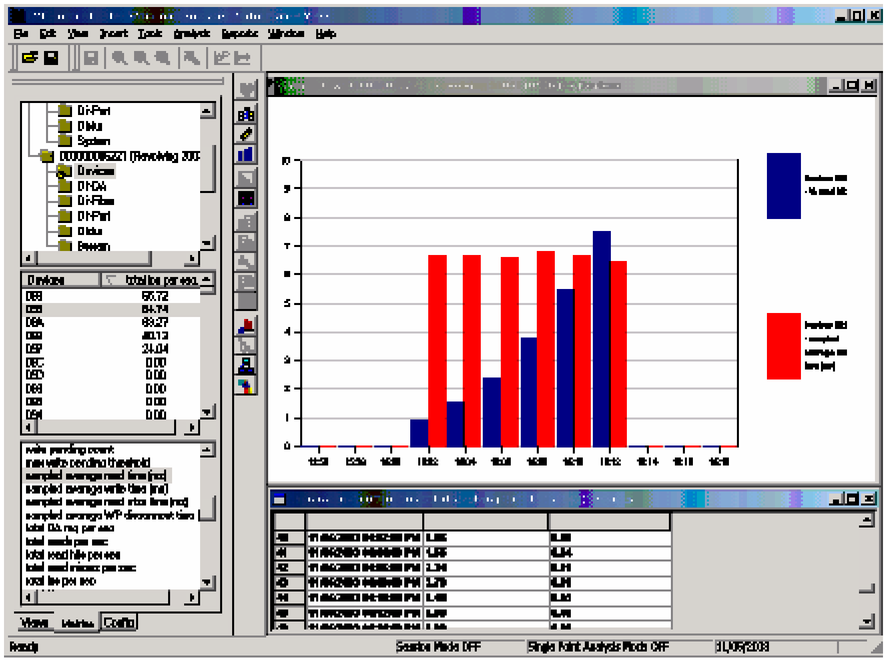

File
Edit
View
Insert
Tools
Analysis
Reports
Window
Help
Devices
total ios per sec
write pending count
max write pending threshold
sampled average read time (ms)
sampled average write time (ms)
sampled average read miss time (ms)
sampled average WP disconnect time
total DA req per sec
total reads per sec
total read hits per sec
total read misses per sec
total ios per sec
Views
Metrics
Config
Ready
Session Mode OFF
Single Point Analysis Mode OFF

Figure 3-41

Host director CPU time for a read hit was calculated back in section 1.2.6.1 (Figure 31).

A spreadsheet is constructed to calculate the average miss service time based on the above formula:

| Average IO Service Time (ms) | Bus Data Transfer Time (ms) | f-e + b-e Time (ms) | Hit Service Time (ms) | % Hit | % Hit * Hit ST (ms) | Miss Service Time (ms) |
|---|---|---|---|---|---|---|
| 6.66 | 0.062 | 6.598 | 0.016 | 0.85 | 0.000136 | 6.716958 |
| 6.64 | 0.062 | 6.578 | 0.016 | 1.55 | 0.000248 | 6.744288 |
| 6.61 | 0.062 | 6.548 | 0.016 | 2.34 | 0.0003744 | 6.767997 |
| 6.81 | 0.062 | 6.748 | 0.016 | 3.76 | 0.0006016 | 7.075435 |
| 6.65 | 0.062 | 6.588 | 0.016 | 5.48 | 0.0008768 | 7.03462 |
| 6.45 | 0.062 | 6.388 | 0.016 | 7.5 | 0.0012 | 6.971676 |

Most of the miss service time is obviously disk time, but since we are working with submilliseconds we need to remember that back-end director time is also included here as well as the second visit to the front-end director to complete the I/O after it was determined that the data is not in cache.

Looking at Figure 3-40 on page 3-35 and Figure 3-41 on page 3-36, we notice that even though the DMSP uses both disks, the single threaded mode of operation for this test does not affect the response time at all. The I/Os are synchronous.

### 3.4.3.2 Multithreaded 8 KB Random Reads

There are four threads active in this test all doing I/Os to the same device. This test will show the benefits of both disk command tagged queuing and the DMSP.

We begin with the I/O rate and the average response times.

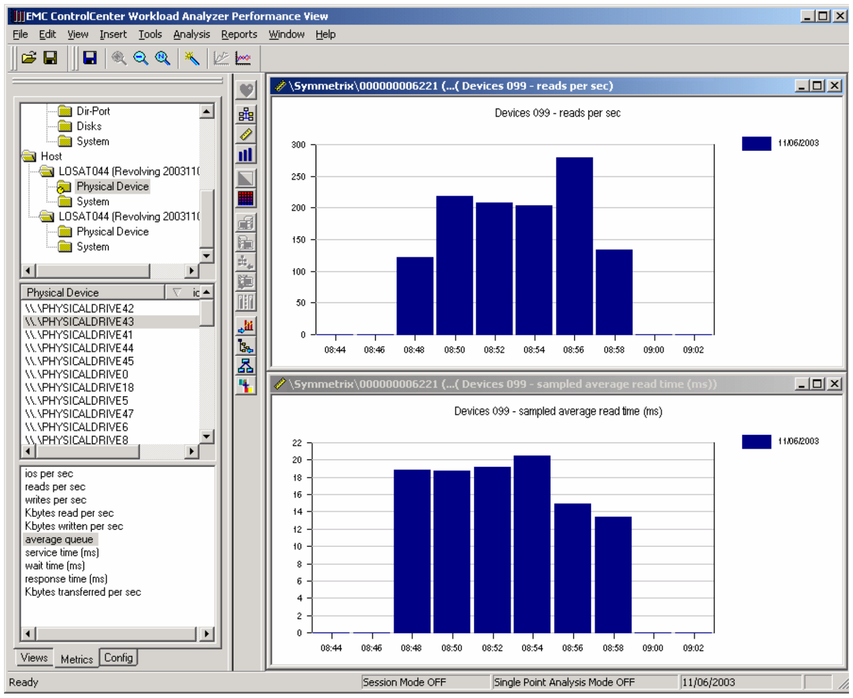


**Figure 3-42**

Since this test uses the same configuration, we see that we obviously did not use 100 percent of our available resources before or else we would not have been able to achieve higher I/O rates with this test.

First, we confirm that we actually have four threads running. We see this from the host perspective by observing the queue length (Figure 3-43).

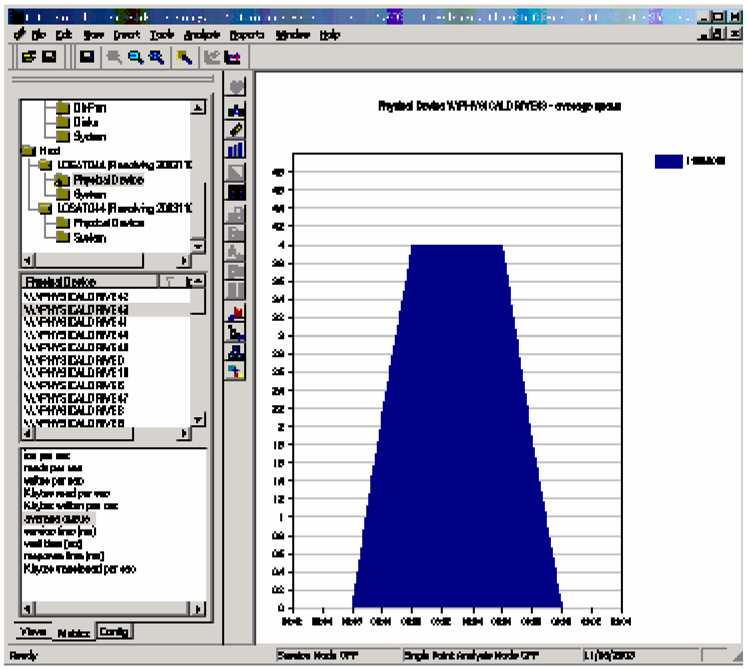


**Figure 3-43**

We can also estimate queue depth the following way:

| I/O Rate | Average Response Time | Total Busy Time | Calculated Queue Depth |
|---:|---:|---:|---:|
| 121.11 | 18.85 | 2282.9235 | 2.28 |
| 218.58 | 18.75 | 4098.375 | 4.10 |
| 208.61 | 19.12 | 3988.6232 | 3.99 |
| 204.11 | 20.48 | 4180.1728 | 4.18 |
| 279.07 | 14.91 | 4160.9337 | 4.16 |
| 133.59 | 13.41 | 1791.4419 | 1.79 |

Since total busy time is calculated in milliseconds, we calculate the total busy time by multiplying the I/O rate by the average response time, and then divide by the elapsed time (1000 milliseconds). This gives us an estimated queue depth value, which as can be seen, for all full time intervals (excluding the edges), the estimate is quite close to the actual as reported in the above table.

This formula will only work for sustained workloads where there is always a queue. In other environments, it could be an indicator of occasional frequent bursts, or it can indicate that generally speaking there were no queues. Occasional bursts could be completely obscured.

### Command Tagged Queuing Affect

We calculated the read-miss service time for an I/O in single-threaded mode to be around 7 milliseconds (table on page 3-37). From what we know now about queues, with a queue depth of 4 we would expect the average service time per I/O to be around 28 milliseconds.

With command tagged queuing the back-end director sends multiple requests to the disk controller. These requests are then re-ordered prior to execution and based on relative physical addressing. As a result, each I/O waits less time.

Figure 3-42 on page 3-37 shows us a maximum response time of around 20 ms Dividing this by the queue depth of four, we see that it took the system less than 5 ms to complete each I/O with a queue effect of 20 ms response time.

Since it took less time to process each individual I/O, we can see that the multithreading actually improved our throughput so instead of around 150 I/Os a second, we climbed to over 200 when only one disk was active, and well over 250 I/Os per second with two disks active.

### DMSP Affect

The Dynamic Mirror Service Policy has two modes: M1/M2 and Interleave. Our system's default was M1/M2. In this mode, the system reads from the data disk for the first 5 minutes before it tries to determine a better policy. In our case, five minutes into the test the DMSP algorithm determined that for the types of I/Os that we were doing (all reads to a single device), the Interleave method was much more effective.

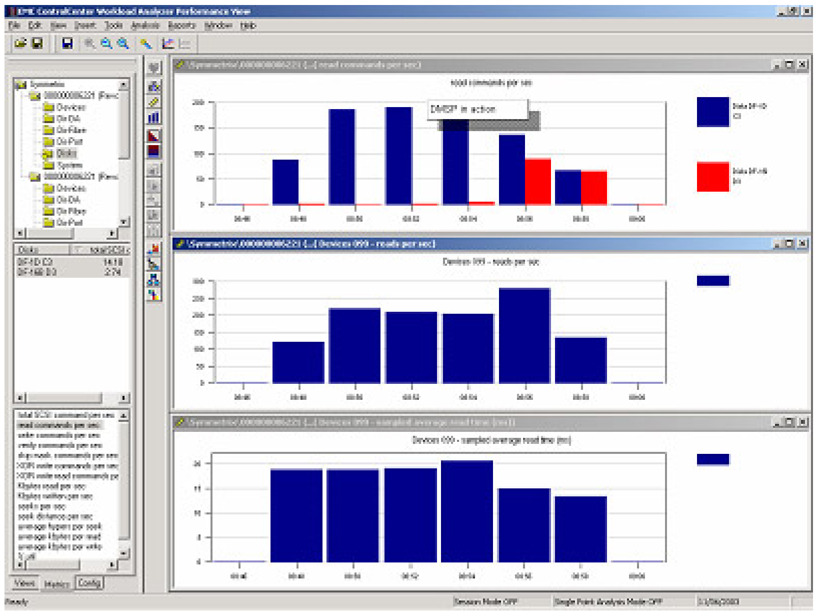

EMC ControlCenter Workload Analyzer Performance View
read commands per sec
DMSP in action
Devices 099 - reads per sec
Devices 099 - sampled average read time (ms)

Figure 3-44

You will notice that for the last full 2 minutes of the test (next to the last bar), we get the full benefit of the DMSP in the form of more I/Os per second and lower I/O response time.

## 3.5 Review

This analysis section described several experiments that used device response time as the focal point for performance analysis. This approach was taken since, as shown, response times can be viewed both at the host and at the Symmetrix system.

Host-based response times are often more closely associated with the end-user application, and application response times are the critical points of pain for users. The characteristics of the I/O streams generated by applications and hosts as well as the number of active applications and threads ultimately determine system performance.

This section analyzed specific I/O streams that were clearly visible since they addressed a handful of devices that could be easily observed. In a busy system where there are hundreds or thousands of devices active, it is difficult to isolate specific devices and follow their lifecycle with sufficient detail. The intent of this section was to familiarize you with the main reasons that slow down response times so that you will know where to look for bottlenecks and queues for improvements.