



Session R03

Diagnosing Windows Server Performance Problems

Gregg McKnight

IBM Distinguished Engineer

xSeries Development Performance

Research Triangle Park, North Carolina

IBM @server xSeries
Technical Conference

Aug. 9 - 13, 2004

Chicago, IL

Introduction - Adjust your thought process

- **Every server environment has a bottleneck!**
 - ▶ Internal
 - Server hardware
 - Server software
 - ▶ External
 - Network
 - Users

- **Bottlenecks do not imply there is a problem**
 - ▶ Unless performance expectations are not being met. E.g.:
 - Poor response time
 - Unable to support additional users

 - ▶ Servers with multiple HW bottlenecks actually deliver optimal price/performance
 - Best case is a server with 100% CPU, Memory, LAN, and Disk utilization
 - Customer is getting maximum value from the server investment

 - ▶ But if customer expects to support a greater workload in the future
 - The system may not manage the workload and a problem will exist!

Introduction

- Windows 2000 or Windows 2003 System Monitor can be used to find just about any internal server bottleneck
 - ▶ The trick is knowing how and where to look?
 - ▶ System Monitor
 - (formally known as Performance Monitor - PerfMon)
 - I like to use the name PerfMon to refer to System Monitor

- There are hundreds of PerfMon counters, this presentation is designed to introduce useful techniques for finding and diagnosing real-world server performance bottlenecks
 - ▶ It does not cover ALL PerfMon counters
 - ▶ Is NOT the only way to find bottlenecks

- For additional Information see:
 - ▶ Overview of Performance Monitoring
 - <http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/windows2000serv/reskit/serverop/part2/sopch05.asp>

Introduction

- It is important to have a consistent plan
 - ▶ Begin with a good overview of how the system is performing
 - Start simple and move through the server components to find the bottleneck(s)
 - ▶ Then gain a deeper understanding of bottlenecked component(s)
 - Is it ?
 - Hardware?
 - ◆ CPU
 - ◆ Memory
 - ◆ Disk
 - ◆ NIC
 - Software?
 - ◆ Application
 - ◆ OS?
 - External?
 - ◆ Users
 - ◆ Network

Introduction

■ How to proceed? Do you trace or monitor real time?

▶ Trace

- Must collect all possible counters needed to diagnose the problem or a new trace will be required

- See appendix for trace instructions
- Best way to diagnose performance problems on your own time
- Downside: Potentially large trace files
- General Bottleneck Detection - Objects to collect - All counters*
 - ◆ Processor
 - ◆ Memory
 - ◆ Physical Disk
 - ◆ Network
 - ◆ System, Processor Queue Length (Only useful counter*)
 - ◆ Process (Useful if you suspect a SW bottleneck, don't trace unless you really need)
 - ◆ Thread (Useful if you suspect a SW bottleneck, don't trace unless you really need)

▶ Monitor Real-time

- Real-time must be done while the problem is occurring
 - Not practical if system workload is dynamic
 - Very flexible - any object or counter can be examined as problem becomes well understood

Section 1 - Diagnosing Server Hardware Bottlenecks

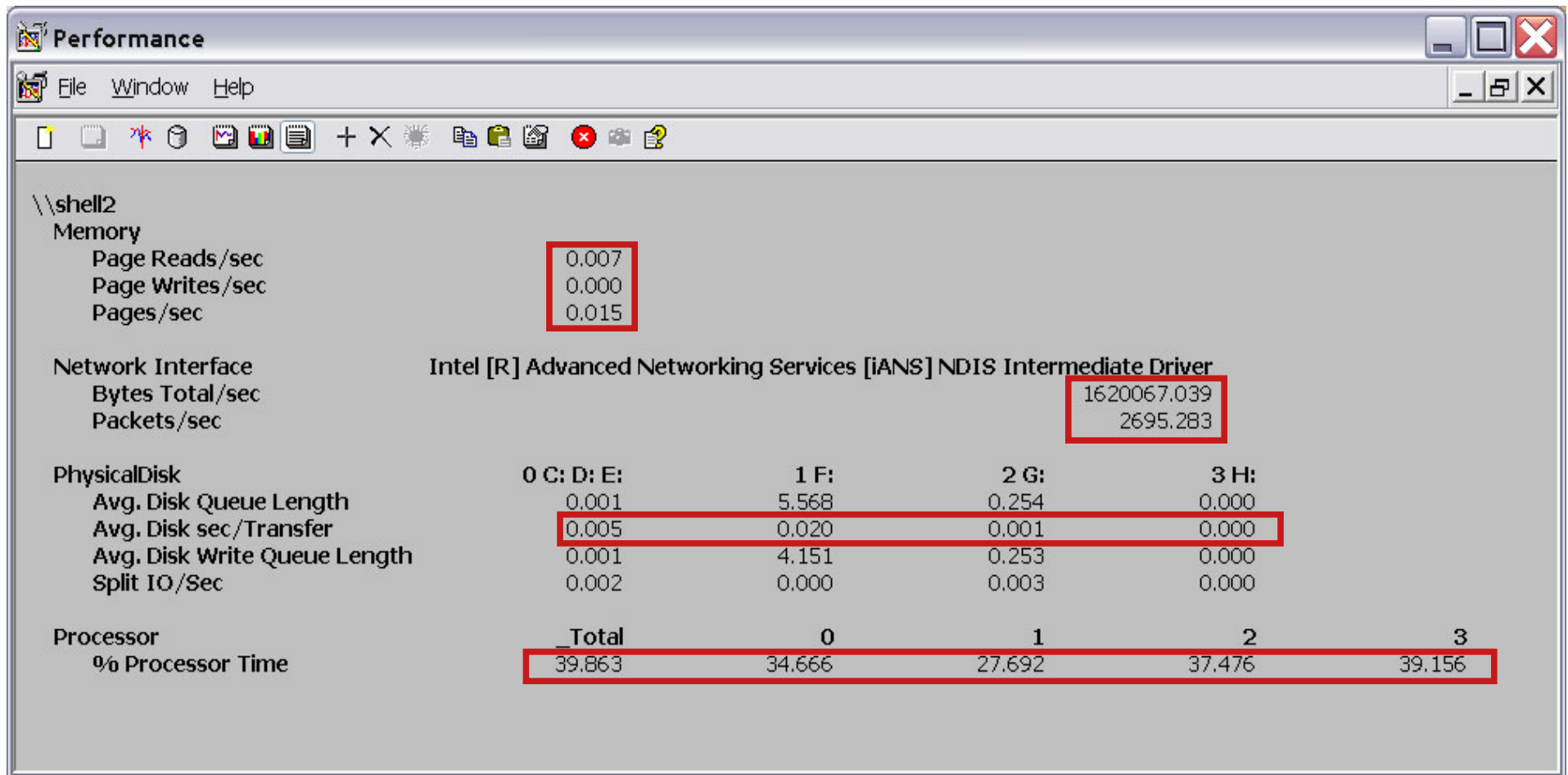
Establishing An Overview

- Develop an overview profile for each HW subsystem
 - ▶ Then decide where to focus

- Keep it simple!
 - ▶ Key Overview Counters are:
 - CPU
 - Object: **Processor**
 - ◆ Counter: **Processor Time** (for each CPU, Not Total)
 - Memory
 - Object: **Memory**
 - ◆ Counters: **Page Reads/sec, Page Writes/sec**
 - Disk
 - Object: **Physical Disk** (for each physical disk, not logical disk)
 - ◆ Counter: **Avg. Disk sec/Transfer**
 - Network
 - Object: **Network Interface**
 - ◆ Counters: **Bytes Total/sec, Packets/sec,**

Establishing An Overview

- I like to use report summary of the busy period for an overview



Processor Overview

■ Object: **Processor**

▶ Counter: **% Processor Time** (for each CPU and Total)

- Monitor each processor individually to spot unbalanced processor utilization problems
 - Only true for Pre-Windows 2003
 - ◆ E.g. If one CPU reaches 100% well before the others,
 - ◆ The server may never reach 100% Total processor utilization
 - ◆ That processor may be the bottleneck
- If any processor is sustaining 80 - 85% utilization for an extended period of time it could be a bottleneck
- Unbalanced processor utilization is usually a software problem
 - **BUT ONLY for Windows 2000 or earlier**
 - Windows 2003 is designed to have unbalanced CPU utilization
 - ◆ This is more efficient because threads will not migrate simply to balance CPU utilization
 - ◆ Introduces processor/memory affinity
- ▶ See Processor Details section
 - if % Processor Time warrants further analysis

Memory Overview

■ Memory

▶ Object: **Memory**

– Key Counters: **Page Reads/sec, Page Writes/sec**

- **Page Reads/sec** are number of DISK READS done for paging
- **Page Writes/sec** are number of DISK WRITES done for paging

▶ **Both represent "hard paging"**

- Other types of paging might not result in disk I/O

▶ **Hard paging is usually sequential I/O**

- This can be several hundred I/Os/Sec/disk before a bottleneck will occur

– If hard paging (total reads and writes) is ~> 200-300 I/Os/Sec/disk of page device

- Then monitor Avg. Disk Sec/Transfer to see if latency to paging drive(s) exceed 15 - 20m Sec (0.015 - 0.020 sec)
- If so, faster paging drives should be evaluated
- BUT if applications support greater memory capacity then add memory first
- Also check to see if paging disk is separate or shared?
 - ◆ Shared will be slower - isolate paging disk if possible

■ Most other memory counters are for virtual memory

- ▶ These cannot be used to diagnose when a server is running low on physical memory
- ▶ Available MBytes useful for understanding physical memory usage

Disk Overview

- **Disk Bottlenecks are the most common bottleneck**
 - ▶ Most configure disks based upon capacity requirements, not performance
 - We purchase largest disks and fewest number of disks (cheapest solution)
 - ▶ But using larger disks places more user data on each disk
 - This causes greater I/O rates to the physical disk and disk bottlenecks to occur

- **Consider disk overhead**
 - ▶ Command = ~1mSec
 - ▶ Seek = ~3 - 4mSec
 - ▶ Rotational latency = ~3 - 4mSec
 - ▶ Transfer data < 1mSec

- **Average random disk I/O is about 8 to 10mSec**

- **Since there are 1000mSec in a second one disk can only do about 125 to 100 disk I/Os per second**
 - ▶ At 8KB per I/O this is only about 800KB/Sec!!!
 - ▶ Disks in commercial servers transfer data at rates much lower than U160 or U320 rates

Disk Overview

- Simplest way to know if disks are a bottleneck is to look at disk service time
 - ▶ Object: **Physical Disk**
 - ▶ Key Counters: **Avg. Disk sec/Transfer, Split I/O/Sec**
 - If disk service time is **much greater than 15 - 20mSec** then the disk is a bottleneck
 - MUST look at physical disk NOT logical disk counters
 - Multiple logical disks can be on one physical disk and counters can be confusing
 - Split I/Os/Sec should not be more than small percentage of I/O E.g. 1%
 - High Split I/Os mean the format allocation size is too small or the array is not aligned
 - Check I/O size by monitoring Avg.. Disk Bytes/Transfer
 - ▶ Monitor disk counters for an extended period of time before determining if disk requires further investigation
 - See disk details section

Network Overview

- Network performance problems can be difficult to diagnose
 - ▶ Network counters in PerfMon only observe data that is to/from the server
 - External network traffic could cause congestion at routers, other servers, clients, etc.
 - ▶ Suspect network problems if:
 - User response time is slow AND:
 - No other obvious system problems
 - Disk latencies are low < 20mSec
 - Sufficient Memory - No/low hard disk paging
 - ◆ Page/Reads and Page/Writes/sec < 100
 - ◆ and Avg. Disk Sec/Transfer < 15 - 20mSec to page device
 - CPU utilization can be high or low
 - ◆ CPU utilization is high if packet sizes are small
 - ◆ Many trips through TCP/IP stack
 - CPU utilization is low if packet sizes are large
 - ◆ Most time spent doing DMA by LAN adapter(s)

Network Overview

- Network Object - Key Counters: **Bytes Total/sec, Packets/sec**
 - ▶ **Chunky Workloads: Bandwidth limited**
 - Image servers, file/web/mail servers, video servers, database warehousing,
 - Bytes Total/Sec should be well below 50-60% of wire speed
 - Wire speed on full duplex adapters is about 1.3 - 1.6x rated speed
 - 1Gbit Enet = 130MB/Sec to 160MB/Sec
 - 100Mbit Enet = 13 - 16MB/Sec
 - ▶ **Chatty Workloads: - High Packet Rates - Packets/Sec maximums:**
 - Message/Chat server, database transaction processing, lots of small files
 - 1Gbit Enet adapter at 50% utilization will drive about 50K - 80K packets/sec
 - 100Mbit Adapter at 50% utilization is about 5K to 8K packets/sec
 - ▶ If Bytes/sec is well below wire speed and Packets/sec is low
 - A network bottleneck could still exist in the external network
 - This will require a LAN Analyzer to diagnose

Bottleneck Overview

- CPU utilization is low, no hard paging, low disk latencies and low LAN Bytes/sec & Packets/sec...
 - ▶ But the server is still under-performing!
 - What now?

- Consider: (This is where it gets hard!)
 - ▶ Software?
 - ▶ External Network?
 - ▶ Users?
 - Usually seen in benchmarks
 - Insufficient client machines are used and low throughput is measured
 - Most servers have the same performance with low to midrange client population

- See details section!

Performance Bottleneck Detection - Examine The Details

Processor Bottlenecks

CPU Utilization Problems - Details

■ Unbalanced CPU utilization

- ▶ Unbalanced CPU utilization in an SMP Server (Windows 2000 or earlier)
- ▶ Can be caused by:
 - Affinity settings?
 - Process pinned to one or more CPUs has outgrown the limited processing power
 - Process is pinned that should float to balance the workload
- ▶ Most other unbalanced CPU utilization problems are caused by software
 - See Software Bottlenecks
- ▶ Sporadic CPU utilization problems can be caused by inefficient software
 - See Software Bottlenecks section

■ Another problem to spot is excessive processor queue length

- ▶ **Object: System Counter: Processor Queue Length**
 - Total count of threads ready to execute queue for all processors
 - Excessive queue length (more than 10 per CPU)
 - ◆ Is good because it means the software is generating work for the processors
 - ◆ Performance can be improved with faster/more CPUs
 - Consistently low queue length indicate software which probably will not scale on SMP systems

Performance Bottleneck Detection - Examine The Details

Memory Subsystem Bottlenecks

Memory Bottlenecks - Details

■ Windows Memory Architecture

- For more detail see: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/memory/base/about_memory_management.asp

▶ Virtual Memory

- Each process is provided 4GB of virtual memory address space
 - But high 2GB is reserved and mapped for kernel
 - ◆ Application must make kernel calls and do so with a 32-bit virtual address address
- This leaves low 2GB for the application process
 - Enterprise enabled applications can steal 1GB from kernel for a total of 3GB virtual memory
 - ◆ 3GB virtual address space for applications is disabled when total physical RAM installed in a computer is equal to or exceeds 16 GB due to the increased size of the kernel from more Page Table pointers.

▶ Physical Memory

- Physical memory is total memory installed in server
- Windows shares the physical memory with **all** processes
 - Hard paging is sustained when total memory demand is greater than physical memory

Memory Bottlenecks - Details

■ Physical Memory - Continued

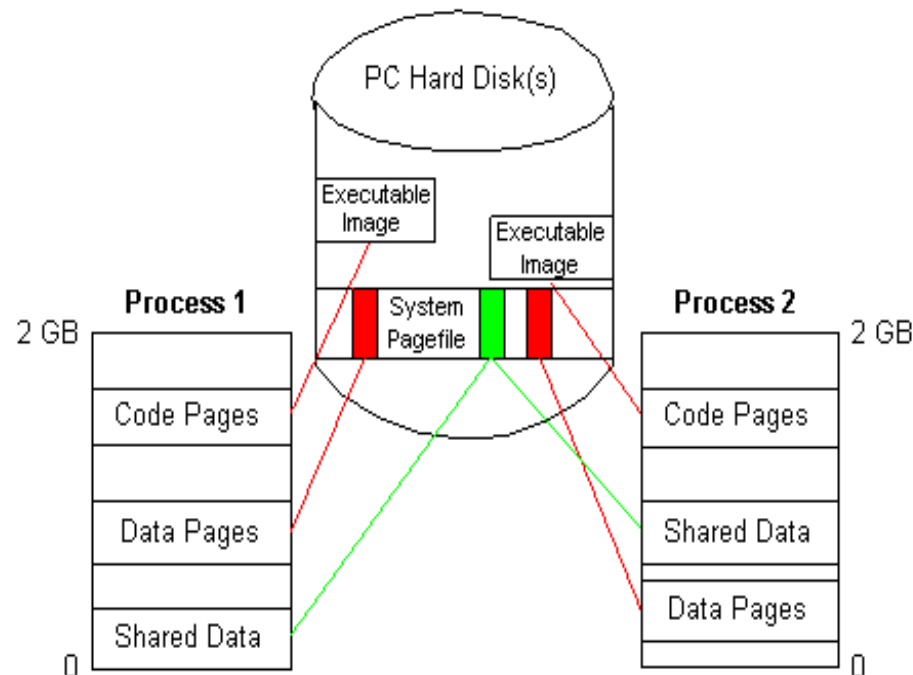
▶ Another technique to increase physical memory size to access or share large files is:

– Memory Mapped Files

- Technique to have fast access to data by using a memory pointer that is backed by paging system
- Paging system does all disk I/O (4KB at a time) so application access is simple and "relatively" quick (I'm not so sure about this, see next chart)
- This technique is used by SAP, NOTES,

◆ See:

- http://msdn.microsoft.com/library/en-us/dngenlib/html/msdn_manamemo.asp?frame=true&_r=1
- http://msdn.microsoft.com/library/default.asp?url=/library/en-us/fileio/base/file_mapping.asp
- <http://support.microsoft.com/default.aspx?scid=http://support.microsoft.com:80/support/kb/articles/Q139/6/09.asp&NoWebContent=1>



Memory Bottlenecks - Details

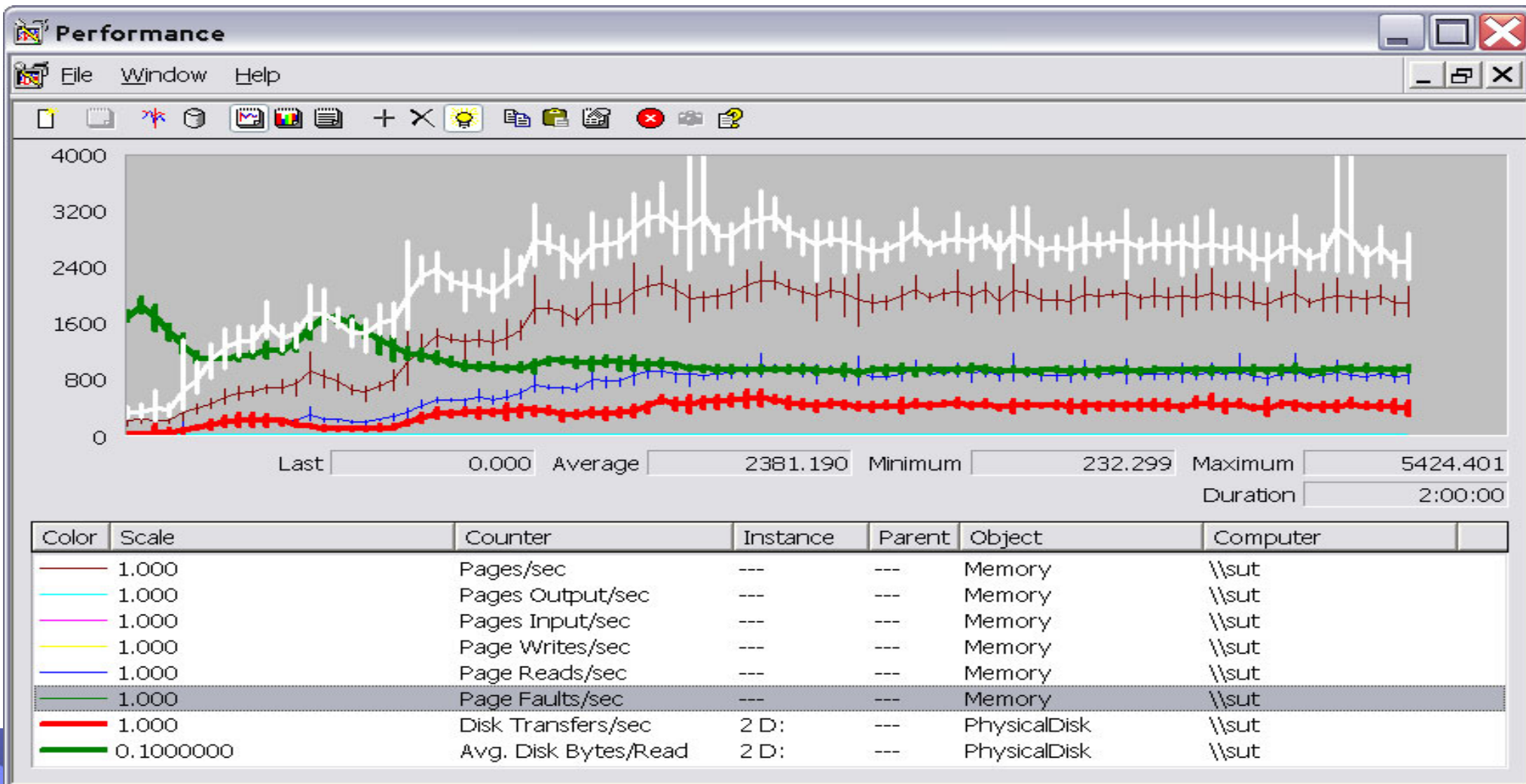
■ Physical Memory - Continued

▶ Memory Mapped Files

- When applications use memory mapped files
 - PerfMon may show very high pages/sec counts
 - Also high Page Reads/Sec and Page Writes/Sec
 - ◆ Because the paging system is "backing the file map"
- Because I/O is usually sequential the rates can be quite high
 - 500 - 1000 Page Inputs/second is not uncommon
- It is hard to know when the application is using memory mapped files
- SAP and Notes memory mapped file I/O
 - But suspect it when very high paging is seen
 - Monitor page disk for latency bottlenecks (> 15 - 20mSec)
 - ◆ If latency is > 15-20mSec increase paging device performance
 - ◆ Newer or faster disks, or greater number of disks in RAID Array
 - ◆ But only up to about Avg. Disk Queue Length / 2
 - Installing more memory may not improve performance!

Memory Bottlenecks - Details Example

- Lotus Notes R5 Mail Server with 10,000 Users
 - ▶ Notes makes heavy use of Memory Mapped Files even with 4GB installed
 - Page Reads/Sec = 800 (~3 pages per disk read)

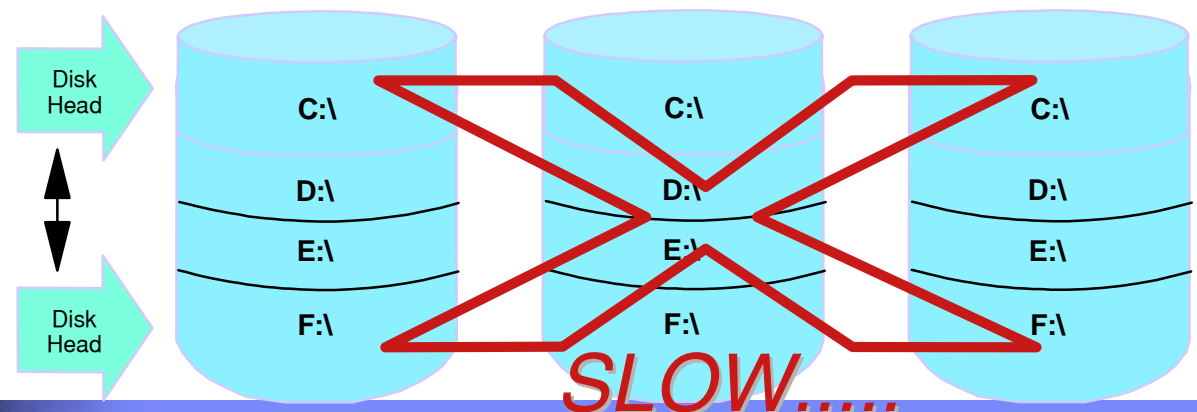


Performance Bottleneck Detection - Examine The Details

Disk Subsystem Bottlenecks

Disk Bottlenecks - Details

- Disk bottlenecks are the most common bottleneck you will find
 - ▶ Worth the time to understand because fixes are usually straight forward
 - ▶ Physical disk counters represent what is happening to the array and are best to use
 - Be careful because an array can be partitioned into LUNs with Array controller
 - ▶ Most common problem you will find is too few disks
 - ▶ Next common occurrence is too many logical disks on the same array
 - This increases seek time and greatly lowers performance
- Disk bottlenecks are confirmed by Avg. Disk Sec/Transfer significantly greater than 15 - 20 mSec
 - ▶ I have seen as high as 300 - 500mSec latency
 - Even up to 30 SECONDS!



Disk Bottlenecks - Examine The Details

- Exchange Server 2000 Example - ~9000 Users
 - ▶ Avg. Disk Queue Length is 4 with 8755 connected users
 - ▶ 1401 active users
 - ▶ This is an I/O bottleneck that does not have long disk latency
 - 4 - 6mSec disk latency
 - Processor utilization is 79% and no additional users can be supported
 - Because I/O is not queued to take advantage of larger amount of disks
 - ▶ Exchange 2000 default config does not drive a deep disk queue length
 - Adding a large number of drives to one array did not help Exchange
 - Best case for default install is a queue depth of 16
 - Typically well below 8
 - ▶ See Config parameter ms-Exch-ESE-Param-Log-Buffers Attribute

Disk Bottlenecks - Exchange Server Example - ~9000 Users

Performance Average

File Window Help

\\SUTEXH
MSEExchangeIS

Active User Count	1401				
Connection Count	8755				

PhysicalDisk

	0 E:	2 F:	3 H:	5 D:	7 G:	
Avg. Disk Queue Length	4.079	4.366	3.045	4.029	0.002	3 - 4
Avg. Disk sec/Transfer	0.006	0.007	0.004	0.006	0.006	

Processor

% Processor Time	_Total	79.419
------------------	--------	--------

System

Processor Queue Length	6
------------------------	---

Performance Maximum

File Window Help

\\SUTEXH
MSEExchangeIS

Active User Count	4973				
Connection Count	9739				

PhysicalDisk

	0 E:	2 F:	3 H:	5 D:	7 G:	
Avg. Disk Queue Length	10.713	11.393	9.361	9.937	0.003	9 - 11
Avg. Disk sec/Transfer	0.013	0.018	0.015	0.014	0.013	

Processor

% Processor Time	_Total	99.583
------------------	--------	--------

System

Processor Queue Length	23
------------------------	----

Disk Bottlenecks - Examine The Details

- Exchange Server 2000 Example - ~11,400 Users
 - ▶ Default Exchange 2000 did not drive a deep disk queue length
 - Default configuration has ms-Exch-ESE-Param-Log-Buffers Attribute set too low (84) for large number of users
 - This causes Log Stalls which pace the disk queue
 - Set to 500 - 9000 depending upon installed memory
 - Each buffer is 512 Bytes
 - ▶ With ms-Exch-ESE-Param-Log-Buffers set to 500
 - Disk Queue increases significantly

Disk Bottlenecks - Exchange Server Example - ~11,400 Users

Average

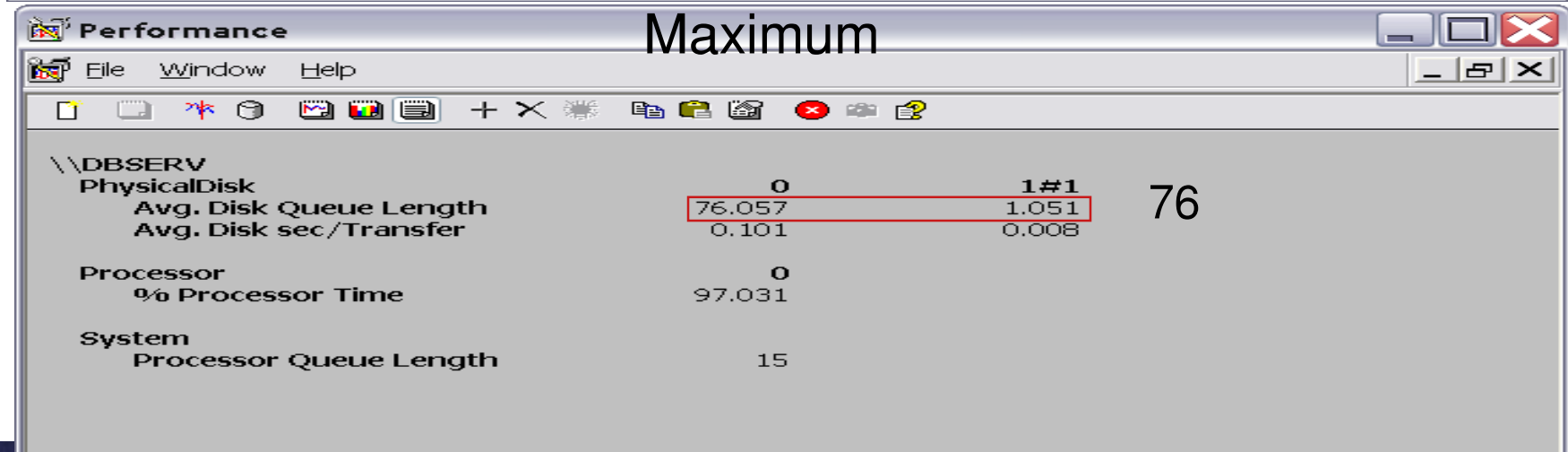
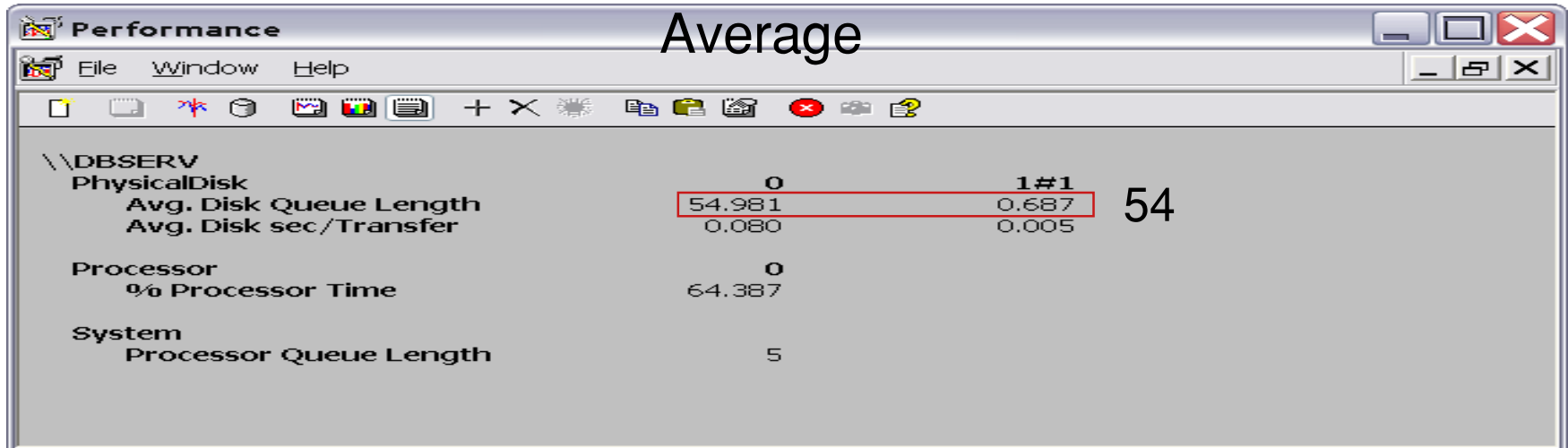
\\SUTEXH		0 C:	1 H: I:	10 G:	6 D:	7 E:	8 F:	6 - 11
MSEExchangeIS								
Connection Count		11411						
PhysicalDisk		0 C:	1 H: I:	10 G:	6 D:	7 E:	8 F:	
Avg. Disk Queue Length		0.007	0.016	6.433	10.980	10.110	6.953	
Avg. Disk sec/Transfer		0.007	0.006	0.008	0.013	0.012	0.010	
Disk Transfers/sec		1.069	2.458	791.838	841.233	830.484	705.185	
Processor		_Total						
% Processor Time		49.458						
System								
Processor Queue Length		2						

Maximum

\\SUTEXH		0 C:	1 H: I:	10 G:	6 D:	7 E:	8 F:	25-32
MSEExchangeIS								
Connection Count		13249						
PhysicalDisk		0 C:	1 H: I:	10 G:	6 D:	7 E:	8 F:	
Avg. Disk Queue Length		0.098	0.124	25.941	27.312	31.515	27.521	
Avg. Disk sec/Transfer		0.045	0.010	0.027	0.031	0.035	0.019	
Disk Transfers/sec		15.650	22.384	1784.028	1755.297	2172.479	1567.123	
Processor		_Total						
% Processor Time		96.774						
System								
Processor Queue Length		28						

Disk Bottleneck - SQL Server 150 Users - 3 Disk RAID-5

- SQL Server default configuration drives disk queue very deep
 - ▶ Utilizes a greater number of disks in an array than just about any application
 - Queue depths can reach 300 or higher



Performance Bottleneck Detection - Examine The Details

Network Adapter Bottlenecks

LAN Bottlenecks - Details

- Bottlenecks in server NIC are usually straight forward to spot
 - ▶ They are either Packet/Sec rate bottlenecks - Chatty
 - ▶ Or Byte/Sec bottlenecks - Chunky

- LAN bottlenecks in the external network are usually more difficult and require network analyzer and network analysis/protocol skills

- The key to detecting server LAN adapter bottlenecks is understanding LAN adapter throughput profiles
 - ▶ Use throughput profiles to understand packet/second rate
 - ▶ Most adapters have a maximum packet rate they sustain for each packet size and throughput then becomes a function of packet size up to wire speed

100Mbit Ethernet - Packet Rate Profile

- 100Mbit Ethernet adapter packet rate profile vs. average frame size
 - ▶ Reference system 2 x 3.06GHz x345 100Mbit LAN
 - ▶ Bytes/Sec / packets/sec = average packet size.

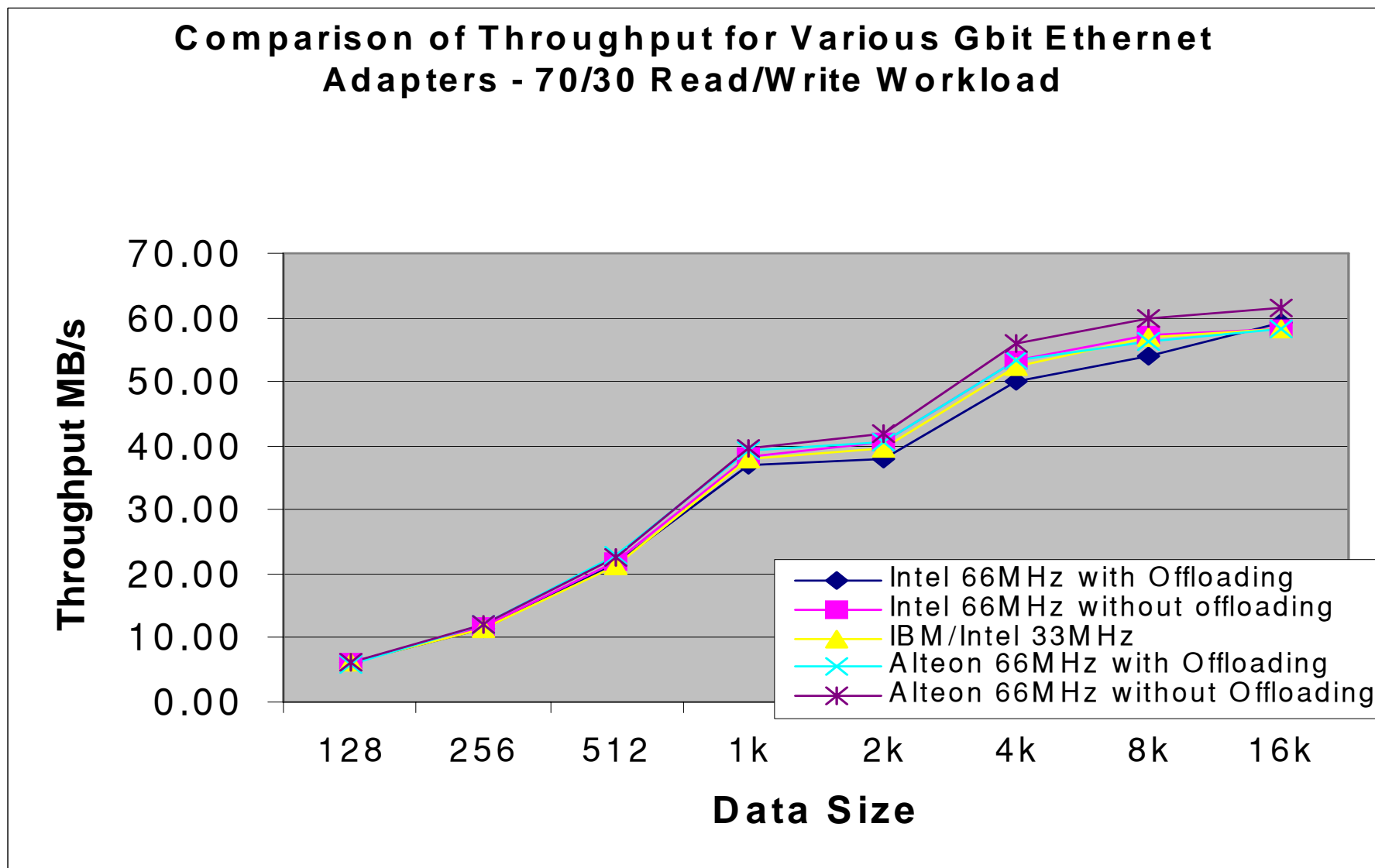
Gbit Ethernet	Packets/Sec	Packets/Sec	Packets/Sec
	100% Client Writes	70/30 Read/Write	100% Client Reads
Packet Size			
64	105,790	103,014	126,553
128	89,939	90,661	90,428
256	55,897	67,044	56,140
512	31,789	39,994	31,788
1024	17,028	21,701	17,024
1460	12,206	15,610	12,202
1480	22,925	31,954	22,932
2k	17,021	23,871	17,024
4k	12,998	17,766	13,001
8k	12,999	18,278	13,002
16k	12,999	18,238	13,004

Latest Generation Gbit Ethernet - Packet Rate Profile

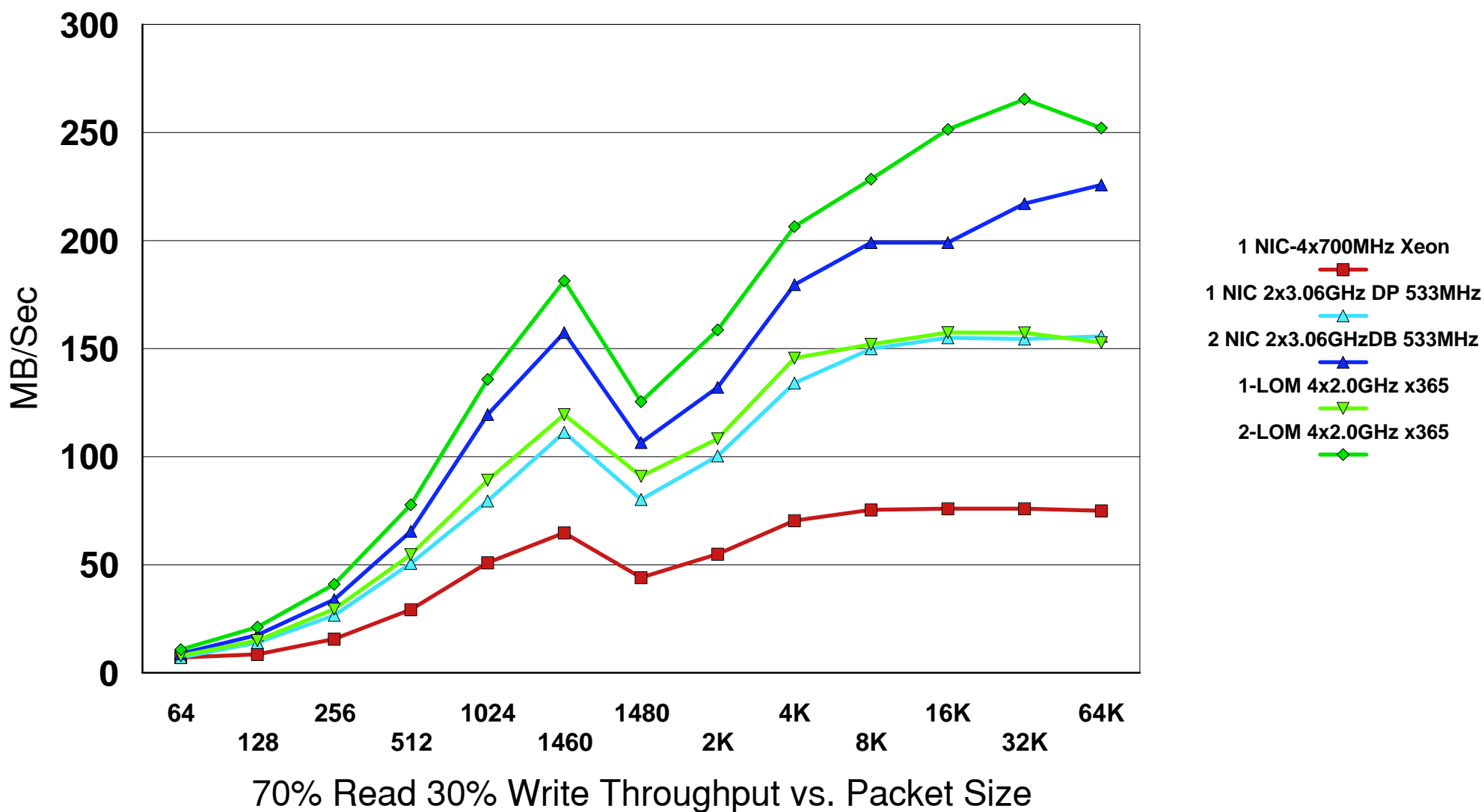
- Gbit Ethernet adapter packet rate profile vs. average frame size
 - ▶ Reference system 4 x 2.0GHz x365 1 Gbit LAN on Motherboard (LOM) active
 - ▶ Bytes/Sec / packets/sec = average packet size.

Gbit Ethernet	Packets/Sec	Packets/Sec	Packets/Sec
	100% Client Writes	70/30 Read/Write	100% Client Reads
Packet Size			
64	125,780	169,218	146,017
128	164,597	161,256	140,588
256	153,649	158,360	137,616
512	135,439	146,796	151,588
1024	114,985	120,586	158,479
1460	107,760	114,046	120,701
1480	143,794	179,335	192,657
2k	122,901	157,194	165,803
4k	124,334	151,746	123,718
8k	102,238	157,377	126,362
16k	115,153	158,683	98,024

First Generation Gbit Ethernet in Xeon PIII Profile



Latest Generation Gbit Ethernet Adapters Throughput Profiles - Windows 2000 - Various Systems



Performance Bottleneck Detection - Examine The Details

Software Bottlenecks

Software Bottleneck - Details

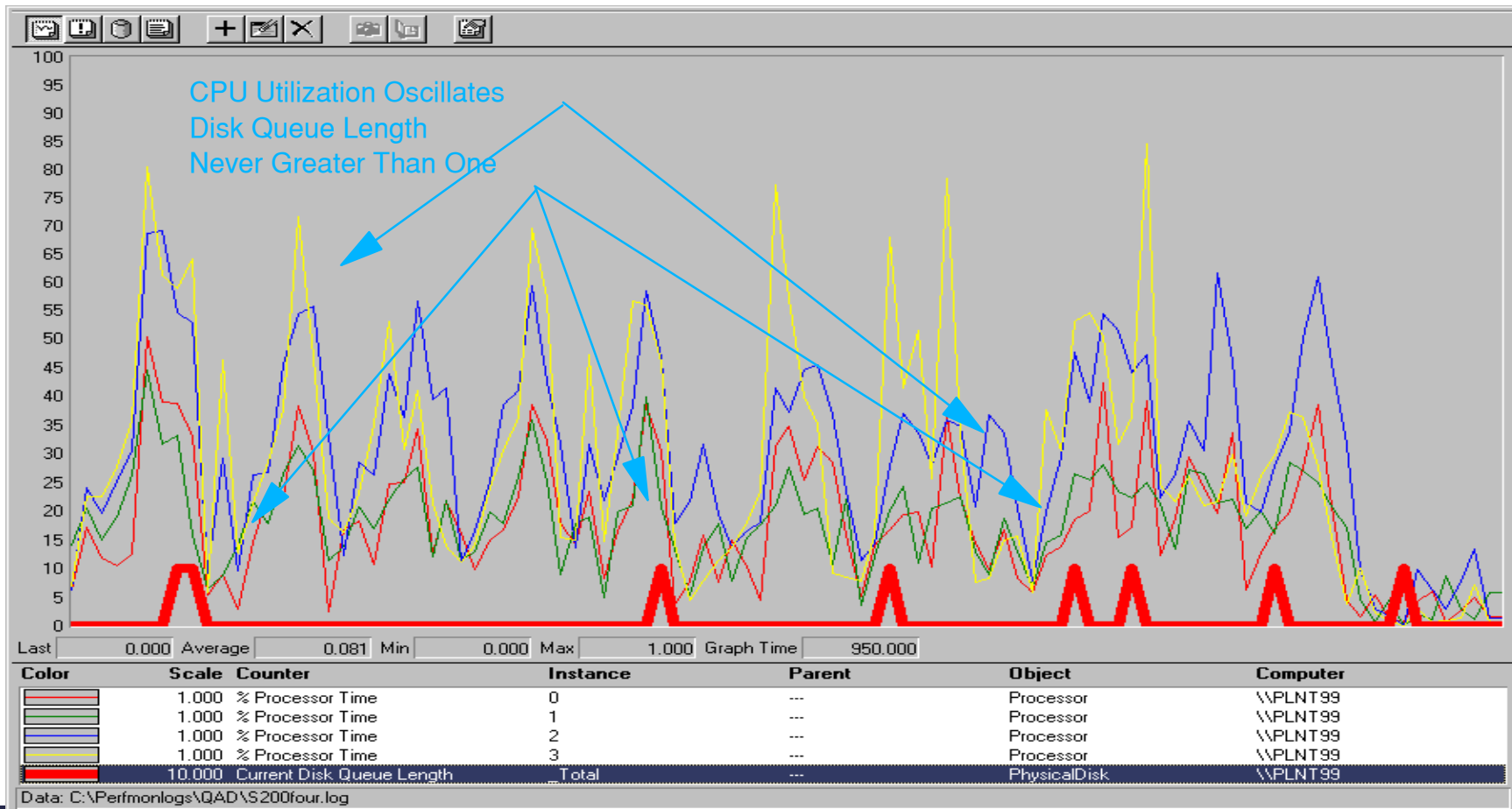
- **SURPRISE! Software does not automatically scale**
 - ▶ Docs from Microsoft worth reading. See:
 - <http://www.microsoft.com/windowsserver2003/docs/Scalarch.doc>
 - <http://www.microsoft.com/windowsserver2003/docs/hiperf.doc>
 - <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vsent7/html/vxconBestPracticesStrategiesForPerformance.asp>

- **Unbalanced CPU utilization - Pre-Windows 2003**

- **Caused by software**
 - ▶ Older, inefficient SW may produce unbalanced CPU utilization
 - Look for a few processors busy while rest are idle or gross unbalanced condition
 - ▶ Double check affinity
 - ▶ For a quick check look at Task Manager/Processes to see what applications are using CPU
 - ▶ Processor utilization caused by older legacy app is usually made worse by poor disk subsystem support
 - Only one or two concurrent disk I/Os generated
 - Makes disk array slow since it cannot keep many disks busy
 - ◆ Disk array is not faster doing one I/O
 - ◆ Disk arrays can only help performance by increasing throughput
 - ◆ But the application has to generate the I/O queue

Software Bottleneck - Old Legacy App - Details

- Application processes data, writes to disk, then issues a new disk read
 - ▶ Then processes data, and issues another I/O...
 - Very little concurrency



Software Bottleneck - Details Example

- Problem - x360 4x1.6GHz runs JAVA middle tier application no faster than x345 2x2.4GHz
- Benchmark results show the two systems have similar throughput
- Customer says x360 does not scale.

Software Bottleneck - Details Example - Big **RED** Flag! - "Same" CPU utilization for both systems!

■ Overview x 335 - 83% Max CPU Utilization

Performance Monitor window for server \\X335-61X. The 'Total' CPU utilization is highlighted with a red box at 83.168%.

Process	Idle	inetinfo	java	java#1	SERVICES	smlogsvc	System	webtool
% Processor Time	67.330	107.481	0.000	165.341	20.407	6.061	17.188	13.920
Processor	Total	0	1	2	3			
% Processor Time	83.168	83.333	82.055	82.813	84.470			

■ Overview x 360 - 49% MAX CPU Utilization

Performance Monitor window for server \\X360-4AX. The 'Total' CPU utilization is highlighted with a red box at 49.799%.

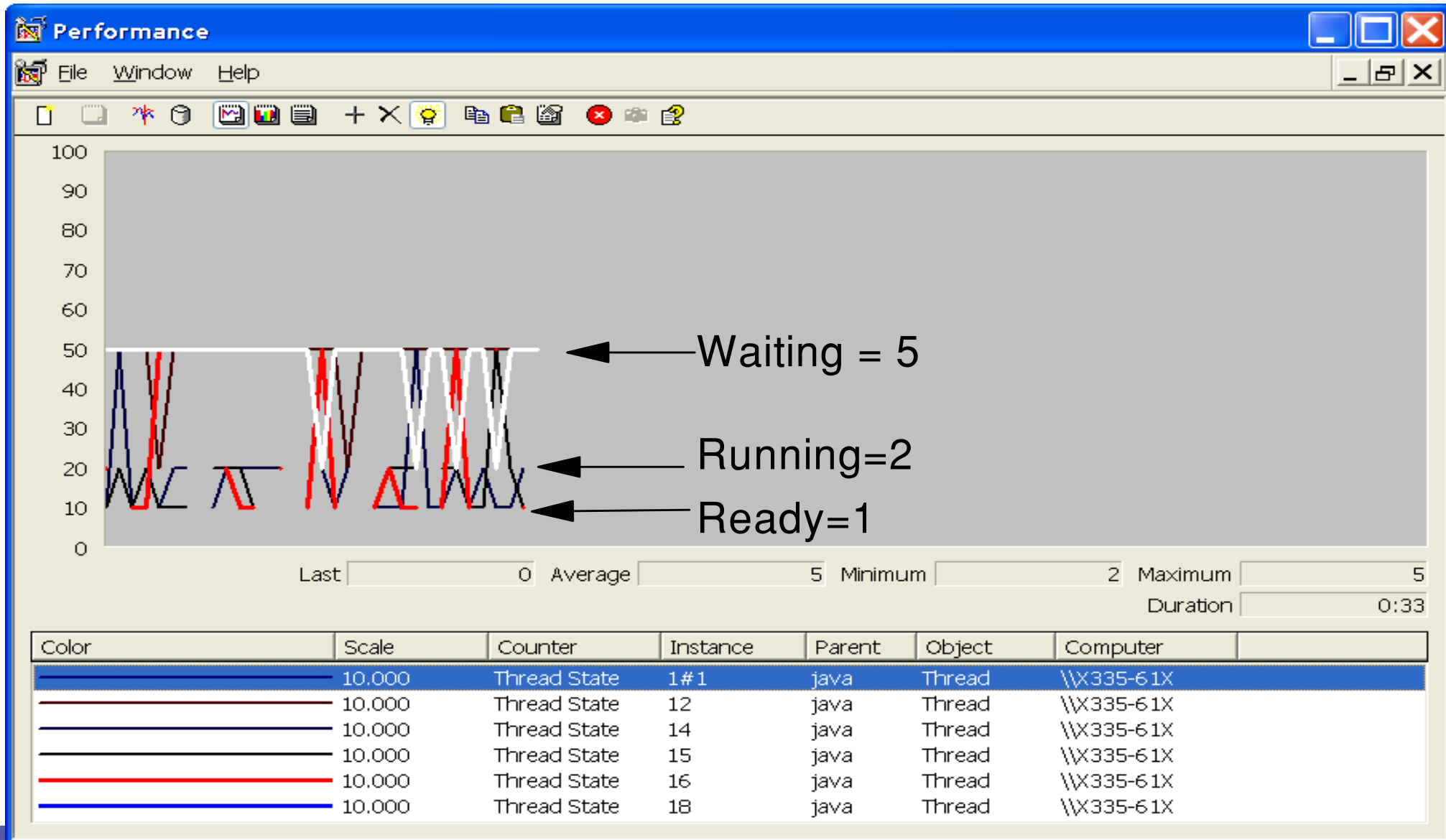
Process	Idle	inetinfo	java	java#1	SERVICES	smlogsvc	System	webtool
% Processor Time	401.606	120.182	181.076	0.000	22.483	9.115	46.181	16.450
Processor	Total	0	1	2	3	4	5	6
% Processor Time	49.799	50.868	43.663	54.297	46.658	49.826	46.094	50.781

Software Bottleneck - Details

- CPU Utilization is about the same for both system
 - ▶ E.g. A 2-way has 1/2 the CPUs of a 4-way so:
 - 2-way at 83% about is doing about the same work as a 4-way at 49%
- Thread Wait Reasons:
 - ▶ Look at Object: Thread, Counters: Thread State, and Thread Wait Reason to diagnose how many threads are running and what the wait reason might be
 - Thread Wait Reason - Is only applicable when the thread is in the Wait state (see Thread State definition object in PerfMon).
 - It is 0 or 7 when the thread is waiting for the Executive,
 - 1 or 8 for a Free Page,
 - 2 or 9 for a Page In, 3 or 10 for a Pool Allocation,
 - 4 or 11 for an Execution Delay,
 - 5 or 12 for a Suspended condition,
 - 6 or 13 for a User Request,
 - 14 for an Event Pair High,
 - 15 for an Event Pair Low,
 - 16 for an LPC Receive,
 - 17 for an LPC Reply,
 - 18 for Virtual Memory,
 - 19 for a Page Out;
 - 20 and higher are not assigned at the time of this writing. Event Pairs are used to communicate with protected subsystems (see Context Switches).

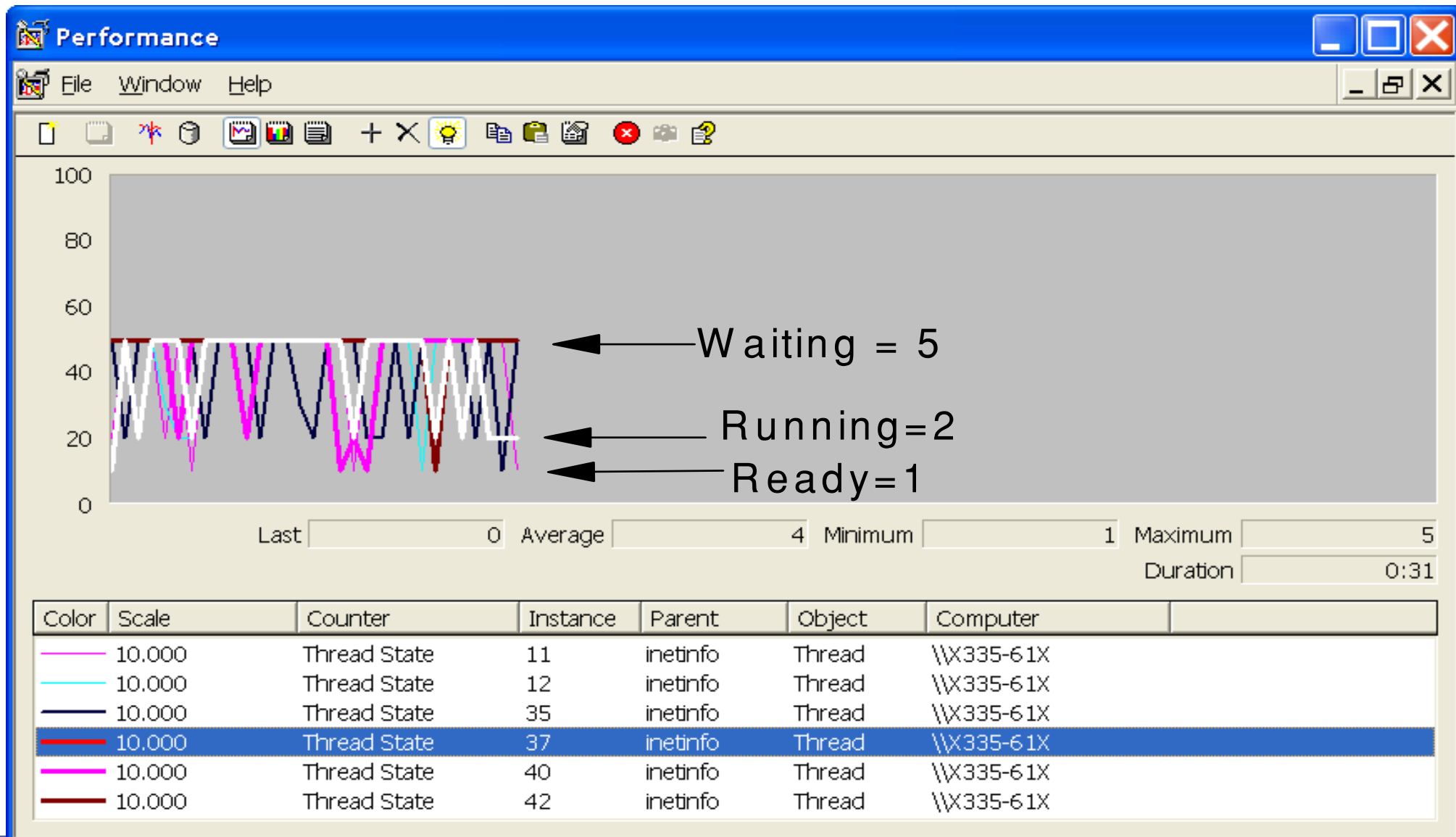
Software Bottleneck - Details

- Java app spending most of the time waiting or ready to run



Software Bottleneck - Details

- IIS app illustrates greater percentage of run time



Performance Bottleneck Detection - Examine The Details

User Bottlenecks

User Bottleneck - Details

- What is a user bottleneck?

- Consider that servers are best measured by capacity
 - ▶ But most think performance is about speed!
 - ▶ Many expect a faster server to do one job or a light number of jobs faster?
 - But this is usually not the case

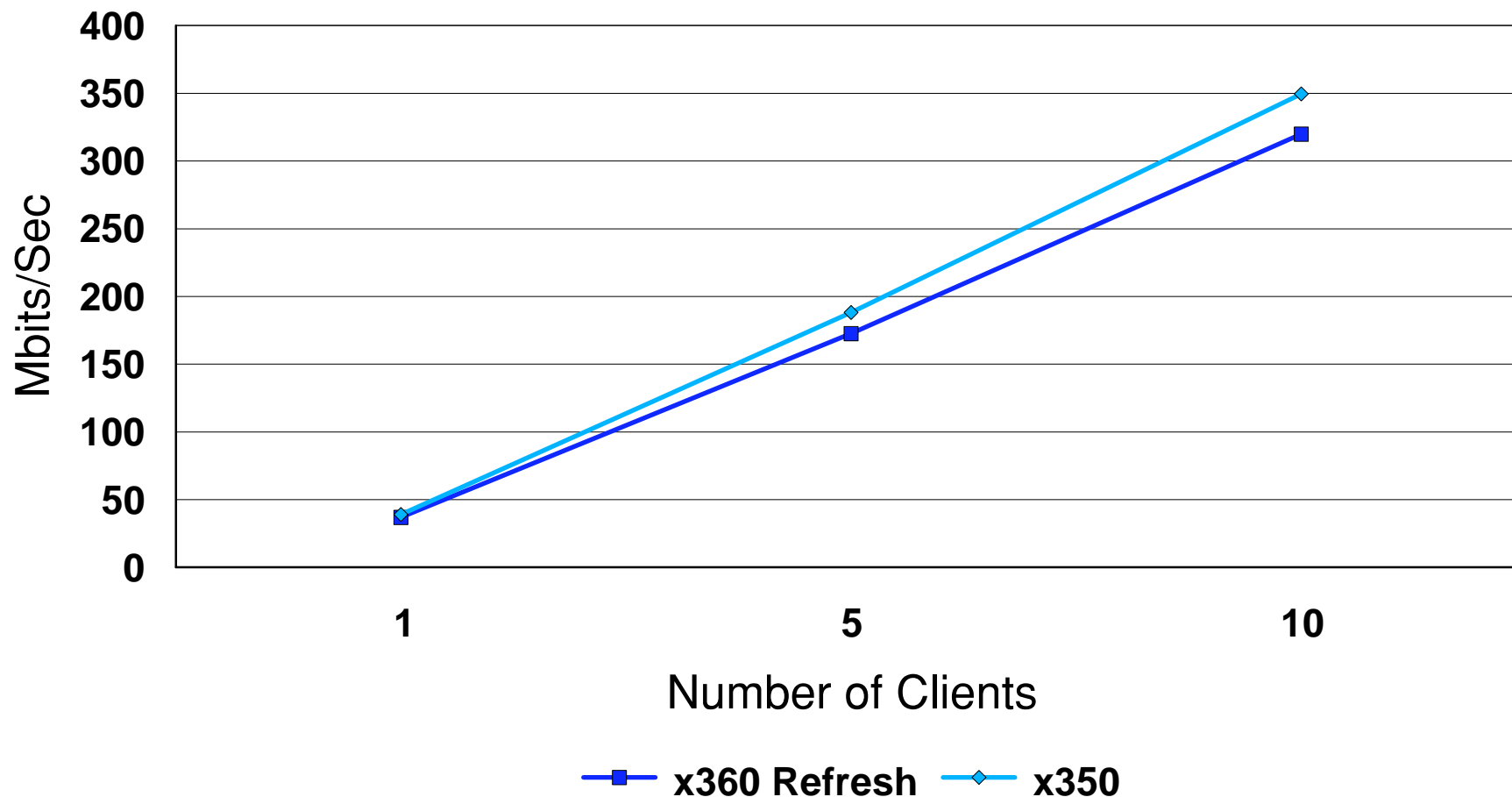
- A "faster" server is usually a "higher capacity" server
 - ▶ It will support more users

Comparing Server Capacity 10 Clients

Windows 2000 File Server Comparison

NetBench 7.01 Disk Mix Benchmark

4 x 1.6GHz 1MB L3 Xeon MP vs. 4 x 900MHz 2MB L2 Xeon

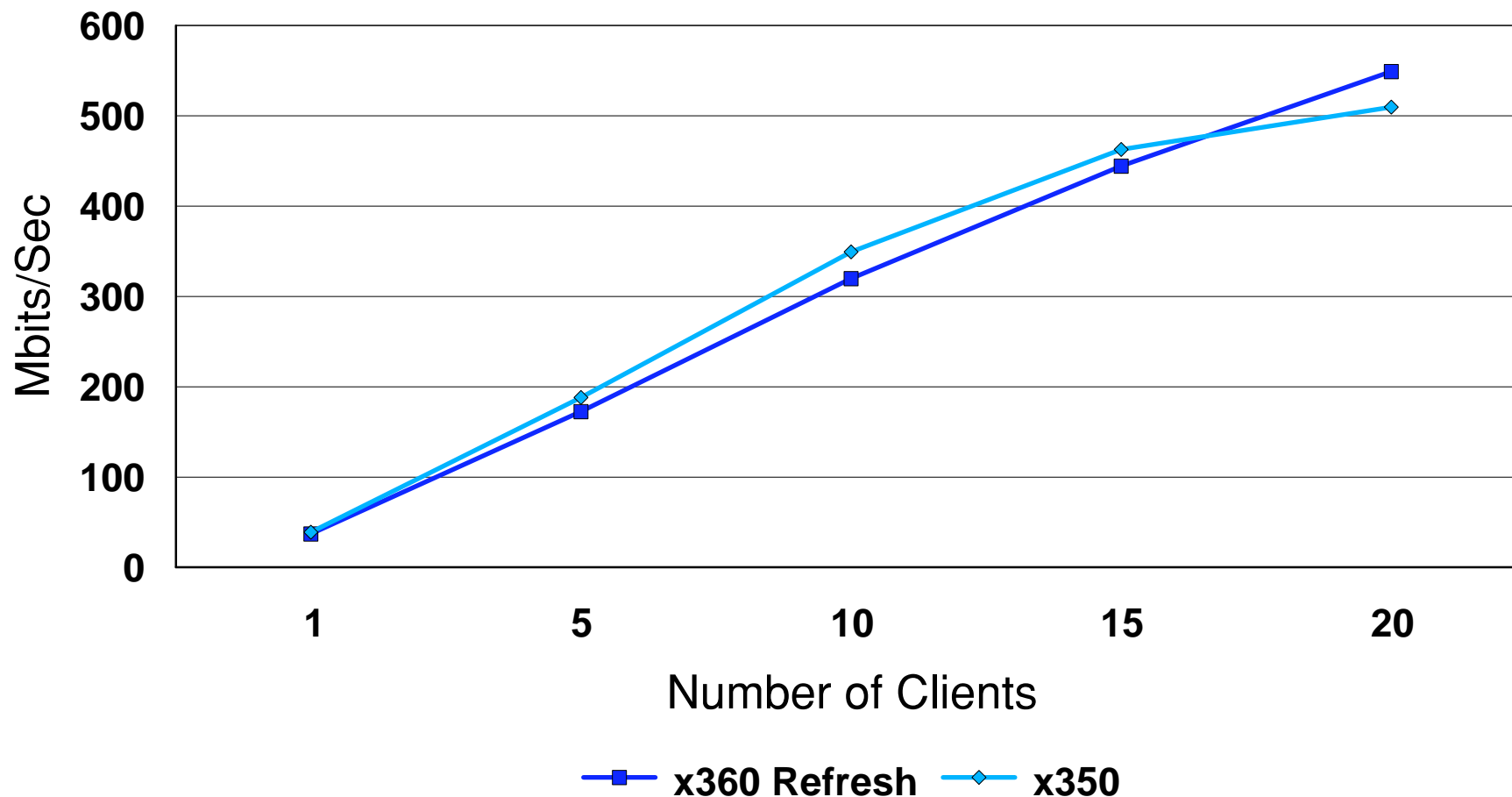


Comparing Server Capacity 20 Clients

Windows 2000 File Server Comparison

NetBench 7.01 Disk Mix Benchmark

4 x 1.6GHz 1MB L3 Xeon MP vs. 4 x 900MHz 2MB L2 Xeon

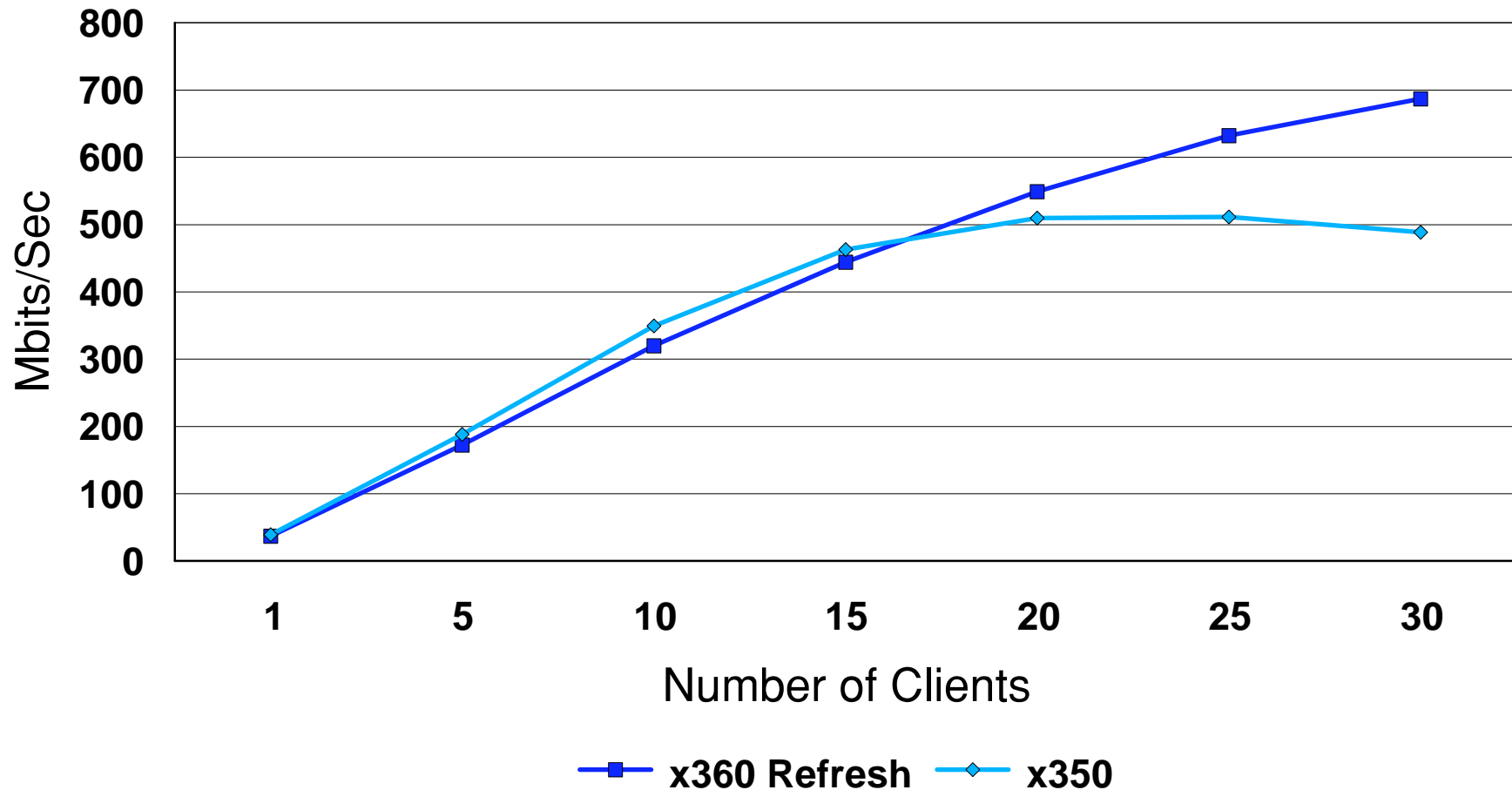


Comparing Server Capacity 30 Clients

Windows 2000 File Server Comparison

NetBench 7.01 Disk Mix Benchmark

4 x 1.6GHz 1MB L3 Xeon MP vs. 4 x 900MHz 2MB L2 Xeon

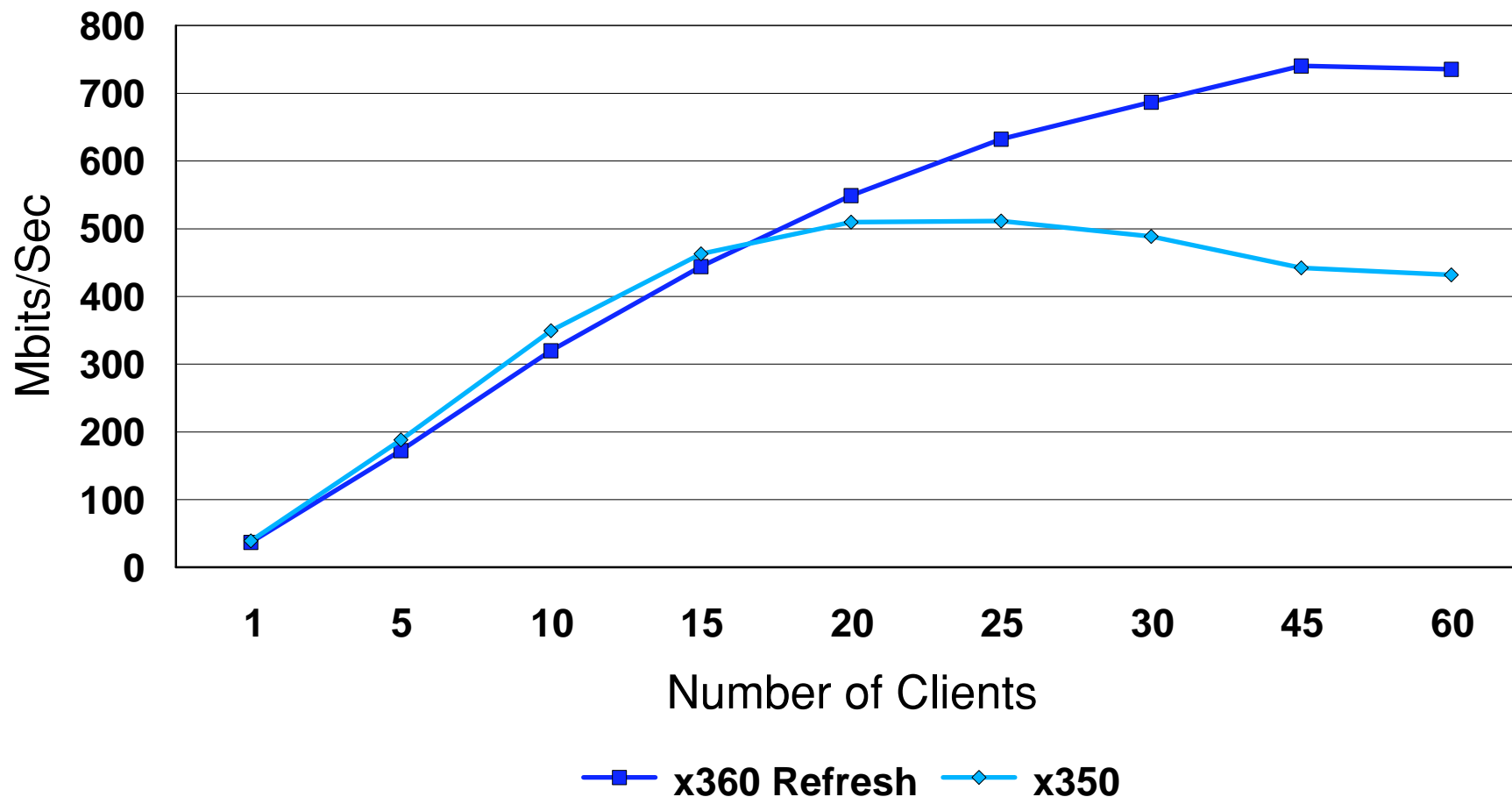


Comparing Server Capacity 40 Clients

Windows 2000 File Server Comparison

NetBench 7.01 Disk Mix Benchmark

4 x 1.6GHz 1MB L3 Xeon MP vs. 4 x 900MHz 2MB L2 Xeon

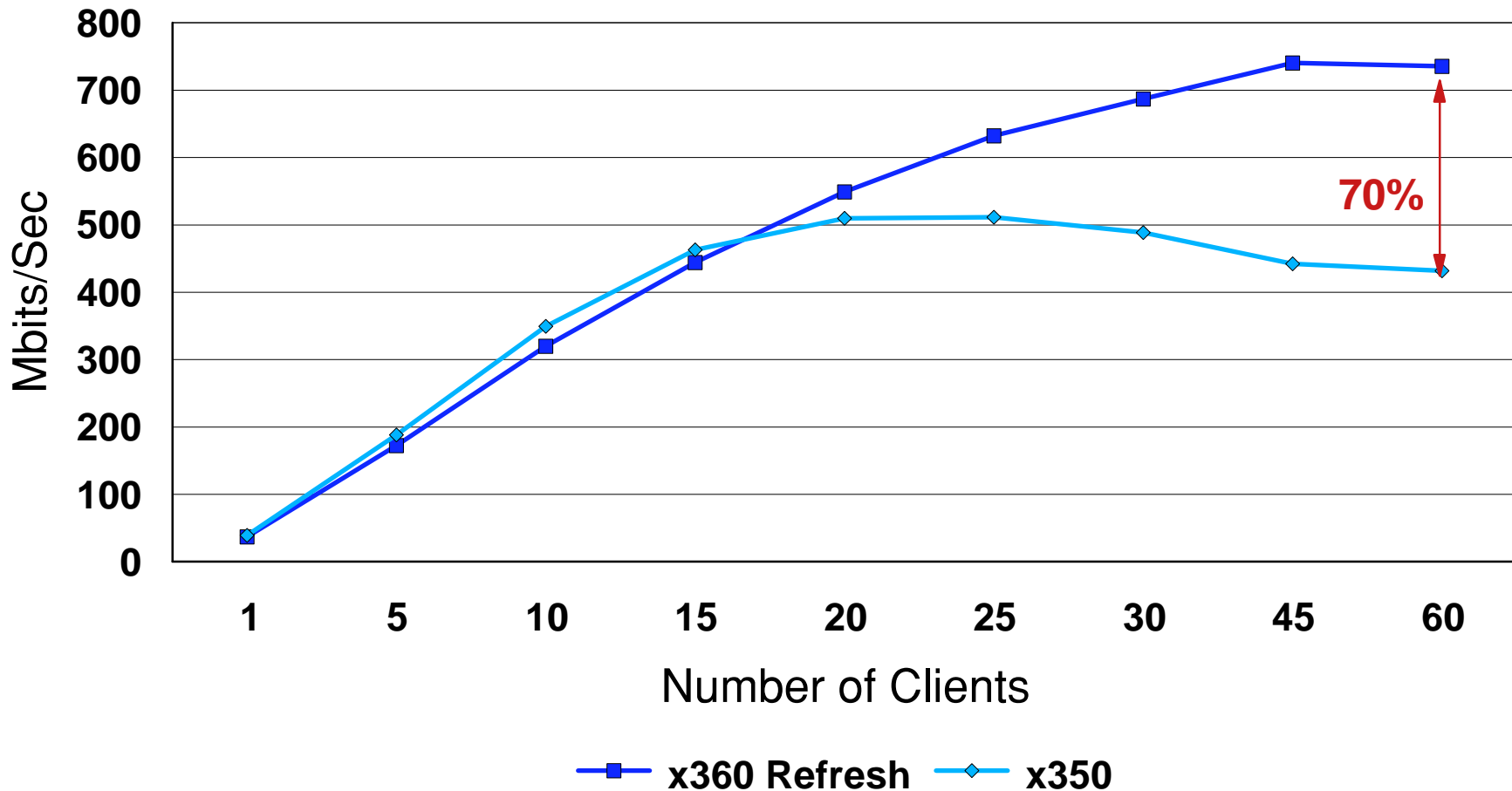


Comparing Server Capacity 60 Clients

Windows 2000 File Server Comparison

NetBench 7.01 Disk Mix Benchmark

4 x 1.6GHz 1MB L3 Xeon MP vs. 4 x 900MHz 2MB L2 Xeon



Customer Shoot-outs ???

- Many customers want to do their own shoot-outs to pre-qualify servers for the corporate approved list
- But these customers don't have 60 systems to use for a benchmark?
- So they cobble together 4 or 5 systems and try to "Benchmark" a 4-way, 8-way and sometimes a 16-way system!!!
- This DOES NOT WORK
 - ▶ For most real server benchmarks there is almost no way to identify true server capacity without a large automated network of clients

Appendix - Using System Monitor

Starting System Monitor

- If you are running Windows 2000 you can start the Performance console as follows:
 - ▶ On the Administrative Tools menu, click Performance.
 - ▶ Or, on the Start menu, click Run, type perfmon.msc, and then click OK.
 - ▶ Or, type perfmon.msc at the Windows command prompt.

- If you are running Windows 2003 you can start the Performance console as follows:
 - ▶ Start, Control Panel, Administrative Tools, Performance
 - ▶ Or, on the Start menu, click Run, type perfmon.msc, and then click OK.
 - ▶ Or, type perfmon.msc at the Windows command prompt.

Instructions to take a System Monitor Trace

(cut/paste into document and enlarge text to read)

You can run the log either directly on the server or remotely.

To create a performance monitor log on the Win2K server do the following:

To open Performance Monitor, click Start, point to Programs, point to Accessories, System Tools, and then click Performance.

To create a counter log

1. Click Performance Logs and Alerts, and then click Counter Logs. Any existing logs will be listed in the details pane. A green icon indicates that a log is running; a red icon indicates that a log has been stopped.

2. Right-click a blank area of the right side details pane, and click New Log Settings.

3. In Name, type name of the log, (enter a name for the trace e.g. customer_name_run)1 and then click OK.

4. On the General tab Click Add Objects

5. At the top: Select Use Local Computer Counter Objects if you are running the log on the server or Select Counters from Computer and point to the correct server name if you are running remotely.

6. Intentionally left blank - Hey. This is IBM ;)

7. Individually Add the following Performance Objects - All counters:

Memory
Network Interface
Physical Disk
Process
Processor
System
Thread

8. Under SAMPLE DATA EVERY INTERVAL: SELECT XX UNITS: SELECT SECONDS

IF THE TRACE IS FOR A PRODUCTION SERVER (PREFERRED) RUNNING FOR A FULL DAY THEN USE XX = 60 SECONDS

One only needs about 8 - 12 hours of a day for the trace (preferably a busy day).

OTHERWISE USE XX = 1 SEC if RUN IS MUCH LESS THAN 1 HOUR

USE 5 SEC for ~1 HOUR RUN

USE 10 SEC for ~2 HOUR RUN

USE 15 SEC for ~3 HOUR RUN

USE 20 SEC for ~4 HOUR RUN...

9. Click Log Files Tab

10. It will show current log file name. Here you can select a drive letter.

MAKE SURE YOU PUT THE LOG ON A DISK WITH FREE SPACE. (At least 500MB.)

PREFERABLY NOT THE DATA DRIVE!) A binary log will zip down about 10X smaller file.

11. Select Log File Type. BINARY FILE

12. Enter comments: Describe the application environment and other relevant data

13. Start numbering should start at 1

14. Click Schedule Tab

Select Manual for both Start and Stop LOG

Please start log before any server activity. Start the test. It is best if you can run the test for at least 15 minutes.

Click OK to Close

When you are ready, right click the little disk under Name on the right side of Performance screen, then select start or stop to begin/end tracing.

The little disk will turn green when trace is active, and red when stopped.

After the trace stops.

It is also helpful to get the system information NFO file. To get this file:

Start/Programs/Accessories/System Tools/System Information/

Click File, Save, Select filename and directory location, File type NFO (System Info File).

In addition to the System Information file you should document the following for later analysis:

(You can edit a text file and add the following information)

1) Number and type of disks in each physical drive.

2) RAID Strategy used E.g. RAID-10 or RAID-5.

Using Windows NT 4.0 PerfMon files with System Monitor

- You can display legacy alert, report, chart, and log settings files in System Monitor by using the following command at the command prompt:
 - ▶ `perfmon.exe settings_file_name`
- When you open one of these settings files, the system temporarily converts the file for use with System Monitor but discards the converted version after the console starts. If you want to save the settings file for permanent use with System Monitor, type the following command:
 - ▶ `perfmon.exe /HTMLFILE:new_file_name settings_file_name`