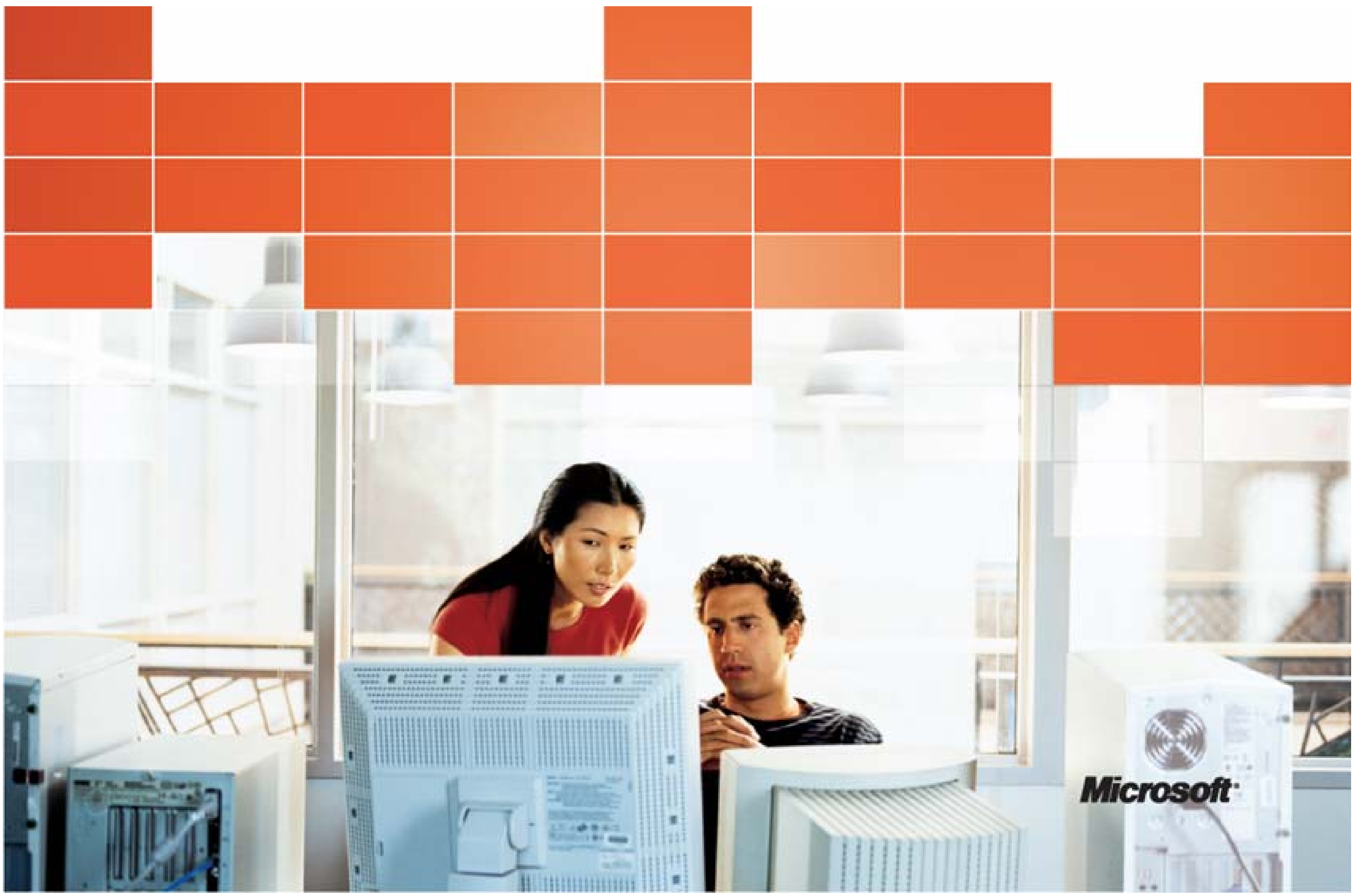
Microsoft Windows Server System

Troubleshooting Exchange Server 2003 Performance

Product Version:	Exchange Server 2003
Reviewed by:	Exchange Product Development
Latest Content:	www.microsoft.com/exchange/library
Author:	Marcus Markiewicz



Troubleshooting Exchange Server 2003 Performance

Marcus Markiewicz

Published: December 2003

Updated: April 2004

Applies To: Exchange Server 2003

Acknowledgments

Project Editor: Olinda Turner

Contributing Writer: Richard Edwards

Contributing Editor: Diane Forsyth

Technical Reviewers: Rob Quimbley, Matt Gossage

Graphic Design: Kristie Smith, Paul Carew

Production: Joe Orzech, Sean Pohtilla

Table of Contents

Introduction

Having one or more subsystems performing below par alters the responses and behaviors of a server. In most cases, this means that the server becomes unresponsive or falls below what would be considered normal operational behavior. This diminished server performance puts the service level agreement (SLA) at risk, which ultimately means that the organization itself can suffer.

A poor performing subsystem also affects other subsystems and even other components of the solution into which the server fits. It is important to be able to pinpoint the cause of the problem without confusing effects from causes. For example, if the server's disk subsystem is causing a major slowdown, mail queues may build up, thereby aggravating the problem and affecting other subsystems, such as increasing the memory consumption.

This technical article provides administrators of Microsoft® Exchange Server 2003 with information on how to isolate performance degradations, and how to use existing tools and procedures for this purpose. The information in this article can also be used to make sure that a server is not degrading over time because of hardware issues or malfunctions. To achieve both of these objectives, the information in this article has been organized into the following sections:

- **Section 1—Getting the Correct Troubleshooting Tools**
  This section lists and describes the performance troubleshooting tools that either come with the Microsoft Windows® operating system or are available on the Internet.
- **Section 2—Identifying the Symptoms of Poor Performance**
  This section identifies which performance counters to use to identify one of the three most common symptoms related to Exchange: rising mail queues, poor client response, and rising Epoxy (ExIPC) queues.
- **Section 3—Isolating the Causes of the Degraded Performance**
  This section describes what can cause degraded Exchange performance, which performance counters to use to troubleshoot each cause, and what you can do to mitigate the effect that each of these causes has on performance.
- **Section 4—Handling Special Exchange Roles and Performance**
  This section focuses on those performance issues related to heavy use of Exchange Server roles, such as front-end servers and public folder servers.
- **Section 5—Handling Multiple Bottlenecks**
  This section deals with those cases where performance degradation stems from multiple causes. It provides an example of a multiple bottleneck situation and shows how to separate and deal with the source of each bottleneck.
- **Section 6—Resources**
  This section lists the Microsoft Knowledge Base articles referenced in this technical article.

> **Note** It is beyond the scope of this article to ensure that a server is healthy, or to diagnose or solve user experience problems. Although performance issues on the server generally translate to a poor user experience (such as slow client response or outages), this is not always true and the reverse does not necessarily hold. Before concluding that a poor user experience results from a poorly performing server, it is important to correlate the user experience issues with observed degradations on the server.

What Is Updated in This Article?

Since the previous version of this article was released, the following modifications were made to Table 6 and Table 17.

- **Table 6 Performance Counters for Database Disks**

  > The maximum values for the PhysicalDisk\Average Disk/sec/Read and PhysicalDisk\Average Disk sec/Write have been changed from 100 ms to 50 ms.

- **Table 17 Performance Counters on the Global Catalog Servers that Indicate Problems**

  > The maximum values for the PhysicalDisk(NTDS Database Disk)\Average Disk/sec/Read and PhysicalDisk(NTDS)\Average Disk sec/Write have been changed from 100 ms to 50 ms.

Terminology

Before reading this technical article, familiarize yourself with the following terms.

Directory Access (DSAccess)
An Exchange 2003 component that provides directory lookup services for components such as Simple Mail Transfer Protocol (SMTP), message transfer agent (MTA), and the Exchange store.

distribution list
A predefined list of mail recipients. Every mail sent to a distribution list causes Exchange to lookup all the members of the list from the global catalog (or, if the data is there, the cache in DSAccess) and expand it in memory so that Exchange can send the mail to the correct recipients.

query-based distribution group
A distribution group that uses a Lightweight Directory Access Protocol (LDAP) query to derive its membership at the time the message is sent. For example, a query-based distribution group might specify an LDAP query such as "Users homed on Exchange server X." When someone sends a message to this query-based distribution group, Exchange executes the associated LDAP query against the Active Directory® directory service, returns a list of all users currently homed on Exchange server X, and then sends the message to that list of users.

global catalog
A server that holds a complete replica of the configuration and schema naming contexts for the forest, a complete replica of the domain naming context where the server is installed, and a partial replica of all other domains in the forest. The global catalog is the central repository for information about objects in the forest.

kernel mode
The processor mode where the core operating system and drivers run. Processes that execute in kernel mode have unbounded access to the processor and all other processes in the system (including user mode processes). Also known as: *privileged mode*.

nonpaged pool
A part of kernel memory that will not be paged to disk.

paged pool
A part of kernel memory that is has either been, or potentially can be, paged out to disk.

remote procedure call (RPC)
A routine that transfers functions and data among computers on a network. It is the mechanism by which the MAPI protocol that is used by Microsoft Outlook® communicates with an Exchange server.

thrashing
The state of a virtual memory system that is spending most of its time swapping pages in and out of memory instead of executing applications.

user mode

The processor mode where the majority of applications run, including almost all Exchange Server 2003 code. User mode applications (processes) have their own address space, and cannot read and write data directly between each other. This prevents potential bugs in an application from taking down the entire system. User mode is ring 3 in the Intel processor architecture. Also known as: *non-privileged mode*.

S E C T I O N 1

Getting the Correct Troubleshooting Tools

When troubleshooting the performance of a server that is running Microsoft® Exchange Server 2003, there are tools that come with the Microsoft Windows® operating system or are available on the Internet that can be of help. The tools listed in Table 1 are useful when diagnosing or preventing a bottleneck, and helping to isolate the problem.

Table 1 Performance Tools for Exchange 2003

Tool name	Description	Install from
Performance (perfmon.msc)	Use for establishing a baseline of performance and for troubleshooting performance issues. **Note** This tool is known as Performance Monitor in Windows 2000 and earlier.	Installed during Windows setup.
Load Simulator (LoadSim) (loadsim.exe)	Use as a benchmarking tool to test the response of servers to mail loads.	Exchange 2003 version: http://go.microsoft.com/fwlink/?linkid=25097
Exchange Stress and Performance (ESP) 2003 (Medusa.exe)	Use to test stress and performance. This tool simulates large numbers of client sessions, by concurrently accessing one or more protocol servers.	Exchange 2003 version: http://go.microsoft.com/fwlink/?linkid=25097
Network Monitor (netmon.exe)	Use to diagnose issues with server connectivity.	Add/Remove Programs ▸ Add/Remove Windows Components
Filemon (filemon.exe)	Use to track I/Os to specific processes and files	http://www.sysinternals.com/

LoadSim and ESP are tools that are helpful when testing systems to ensure the health of the systems before going into production. You can use the other tools to help diagnose bottlenecks in production servers.

Performance

Performance is a Microsoft Management Console (MMC) snap-in that enables monitoring numerous subsystems and software over time. It provides a common infrastructure for reporting data based upon performance counters. These counters are organized hierarchically, as follows, by object, counter and (optionally) instance:

- **Performance Object** This is the part of the computer being monitored. Some of the most commonly used objects are Processor, Memory, and PhysicalDisk. Installing Exchange 2003 adds new objects, such as MSExchangeIS, to the performance object list.
- **Counters** The counters available for a performance object are the parts of the object you can monitor. For example, you can monitor the available bytes, kilobytes, and megabytes of memory, as well as the page faults per second or total pages per second, for a Memory object.
- **Instances (optional)** There may be multiple objects or counters to monitor on the computer. For example, when looking at counters under the Processor object on a multiple processor computer, you see as many instances as there are processors on that computer. You can choose to monitor only a specific processor or all processors.

There are many performance counters covered in this article. Using the Performance snap-in, you can easily glance at the current values of these counters and verify the current status of each counter. However, possibly the most interesting feature is the ability to log any number of counters, objects, and instances across any number of machines to either a file or an SQL database.

The performance counters referenced in this article have the following format:

Performance Object(Instance)\Counter

> **Note** The instance is not a requirement. For example, the performance counter, PhysicalDisk\% Disk Time, does not have a specific instance.

Knowing When to Monitor Performance

Knowing which counters to monitor using the Performance snap-in is only part of monitoring performance. You also need to know when to monitor these counters. A server has many states and, by monitoring each of these states, you derive an understanding on what is normal server operation and what is not.

When you bring an Exchange server online, it takes some time until Exchange fills the caches properly and populates all memory structures. This state is called the transient state. The length of the transient state varies based upon the size of the server (number of processors, memory, and other factors) and the load applied to it. Typically, the transient state varies from 2 hours to 1 per day.

After the transient state is over, the server enters a state where it should exhibit a consistent load with a predictable variance. This state is called the stationary state, and should hold indefinitely unless the load changes or external factors take place, such as backups, restores, database maintenance, and so on.

To analyze a server, it is important to log the performance of server with as many data points as possible, so that the sampling rate is faster than the effect being measured. If possible, a cyclical performance degradation should be logged with data points before, during, and after it took place—the entire cycle. In addition, it is recommended to use a sampling resolution of at least one minute or less to capture performance issues. For example, disk spikes that subside in 5 minutes can be tracked with 30-second Performance logs.

If there is no evident degradation in specific periods of time, logging during an extended period such as one entire day cycle in the life of a server is generally very revealing. It is important to understand how the server reacts as users start using the server in the morning, during peaks hours, and at the end of the day. It is also important to understand how the server reacts during database maintenance and backup cycles. Sometimes, an ill-conceived backup schedule may force the database maintenance not to finish properly, which affects the server performance later.

Load Simulator (LoadSim)

Load Simulator or LoadSim 2003 is a tool that simulates the performance load of MAPI clients. It is useful tool for administrators who are sizing servers and validating a deployment plan. Specifically, LoadSim helps you determine if each of your servers can handle the load that you intend them to carry. Another use for LoadSim is to help validate the overall solution.

However, when it comes to sizing servers completely, LoadSim does not account for all factors. LoadSim does not simulate the following factors that can affect your server capacity planning:

- Incoming unsolicited commercial e-mail (also known as spam) from the Internet
- Incoming Simple Mail Transfer Protocol (SMTP) mail flow from the Internet or other sites within your organization
- Use of non-MAPI protocols for account access, such as Post Office Protocol version 3 (POP3) and Internet Message Access Protocol version 4rev1 (IMAP4)
- Use of mobile devices
- Public folder utilization

In addition, LoadSim does not give a complete picture with regards to user experience, and its results should not be interpreted in that aspect.

For more information about LoadSim 2003, read the file, Using LoadSim 2003.doc, that comes with the tool.

Exchange Stress and Performance

You can use Exchange Stress and Performance (ESP) to simulate arbitrarily several client sessions concurrently accessing one or more Exchange 2003 servers.

ESP provides modules that simulate client sessions over the following Internet protocols/APIs:

- WebDAV (for Microsoft Outlook® Web Access)
- IMAP4
- Lightweight Directory Access Protocol (LDAP)
- OLE DB
- Network News Transfer Protocol (NNTP)
- POP3
- SMTP
- Outlook Mobile Access Sync
- Outlook Mobile Access Browse

Therefore, ESP should be considered as a tool that is similar to LoadSim, but is used when validating deployments that use mobility features and Internet protocols that LoadSim does not cover.

For more information about ESP, read the documentation that comes with the tool.

Network Monitor

Network Monitor enables you to detect and troubleshoot problems on LANs. Using Network Monitor, you can:

- Identify network traffic patterns and network problems. For example, you can locate client-to-server connection problems, find a computer that makes a disproportionate number of work requests, and identify unauthorized users on your network.
- Capture frames (packets) directly from the network.
- Display, filter, save, and print the captured frames.

For instructions for about using Network Monitor to troubleshoot performance, see the "Troubleshooting Network Utilization" section later in this article. For more information about Network Monitor, see the following Microsoft Knowledge Base articles:

- Microsoft Knowledge Base article 294818, "Frequently Asked Questions About Network Monitor" (http://go.microsoft.com/fwlink/?linkid=3052&kbid=294818)
- Microsoft Knowledge Base article 148942, "How to Capture Network Traffic with Network Monitor" (http://go.microsoft.com/fwlink/?linkid=3052&kbid=148942)

Filemon

Filemon is a third-party tool available from Sysinternals (http://www.sysinternals.com/). This tool monitors and displays file system activity on a system in real-time. Its advanced capabilities make it a powerful tool for exploring the way Windows works, seeing how applications use files and DLLs, or assessing problems in system or application file configurations. The time stamping feature of Filemon precisely indicates when every open, read, write, or delete occurs, and its status column indicates the outcome. Filemon begins monitoring when you start it, and its output window can be saved to a file for off-line viewing. It has full search capability and filters.

For more information on the Sysinternals Filemon tool, see the "Troubleshooting I/O Transfers" section later in this article and the Sysinternals site.

> **Note** This third-party contact information is provided to help you find the technical support you need. This contact information is subject to change without notice. Microsoft in no way guarantees the accuracy of this third-party contact information.

SECTION 2

Identifying the Symptoms of Poor Performance

With the correct troubleshooting tools, you can begin to diagnose where Exchange 2003 performance has been degraded. When a server begins to perform poorly, it is often because a particular subsystem has begun to underperform. For a mail server such as Exchange, performance degradation always has the symptoms of rising mail queues, poor client response, or rising Epoxy queues.

Identifying Rising Mail Queues

In Exchange Server 2003, Simple Mail Transport Protocol (SMTP) is the default transport for messages across servers and to the Internet. All mail that is not being sent from senders and recipients homed on the same server goes through SMTP.

The path with which a message goes generally consists of these steps:

1. The message is submitted using MAPI or SMTP.
2. The message is categorized, which means the Exchange Server consults Active Directory for information about its recipients.
3. The message is routed, which means the Exchange Server decides if the message should be delivered locally (the recipient is homed on this server) or to which server should this message be sent (to the Internet, another server, other domain, and so on).
4. SMTP either delivers the mail locally or queues it for remote transfer.

In most cases, a buildup of messages in the Local Delivery queue indicates a performance issue or outages on the server, because the server can no longer deliver the incoming mail in a timely manner. This hold up can come from slowness in consulting Active Directory or in handing messages off for local delivery or SMTP. It can also come from databases being dismounted.

A rise in the remote queue length means that mail is not being sent to other servers. This failure to send mail can be explained by outages or performance issues with the network or remote servers. Those outages or performance issues are causing the network or remote servers from receiving the mail efficiently.

To determine whether message queues are rising, look at the counters listed in Table 2. These counters show different mail queues in Exchange and what their expected values are. If the counters do not match the expected values, performance is suffering from a buildup in message queues.

Table 2 Performance Counters for Mail Queues

Mail queue\Counter	Expectations
SMTP Server\Local Queue Length Indicates the number of messages in the SMTP queue for local delivery.	• The maximum value should be less than 1,000. • The queue should remain steady near its average with small variance.
SMTP Server\Remote Queue Length Indicates the number of messages in the SMTP queue for remote delivery.	• The maximum value should be less than 1,000. • The queue should remain steady near its average with small variance.
SMTP Server\Categorizer Queue Length Indicates the number of messages in the SMTP queue for DS attribute searches	• The maximum value should be less than 10.
MSExchangeIS Mailbox\Send Queue Size Indicates the number of messages in the mailbox store's send queue.	• The send queue size should be less than 500 at all times.
MSExchangeIS Mailbox\Receive Queue Size Indicates the number of messages in the mailbox store's receive queue.	• The send queue size should be less than 500 at all times.
MSExchangeIS Public\Send Queue Size Indicates the number of messages in the public folder's send queue.	• In a server with no mail-enabled public folders, it should be below 10. Otherwise, it should be below 500 at all times.
MSExchangeIS Public\Receive Queue Size Indicates the number of messages in the public folder's receive queue.	• It should be below 500 at all times.

Example of Monitoring Mail Queues

In this example, you see increasing mail queue length as a symptom of increasing server load during a morning peak. As shown in Figure 1, the peak utilization time for this Exchange server starts at 9:00 A.M. During this peak utilization, there is consistently an **MSExchangeIS Mailbox\Send Queue Size** counter in the store averaging 200 and the **SMTP Server\Local Queue Length** counter grows to over 15,000 messages until the peak time finishes.

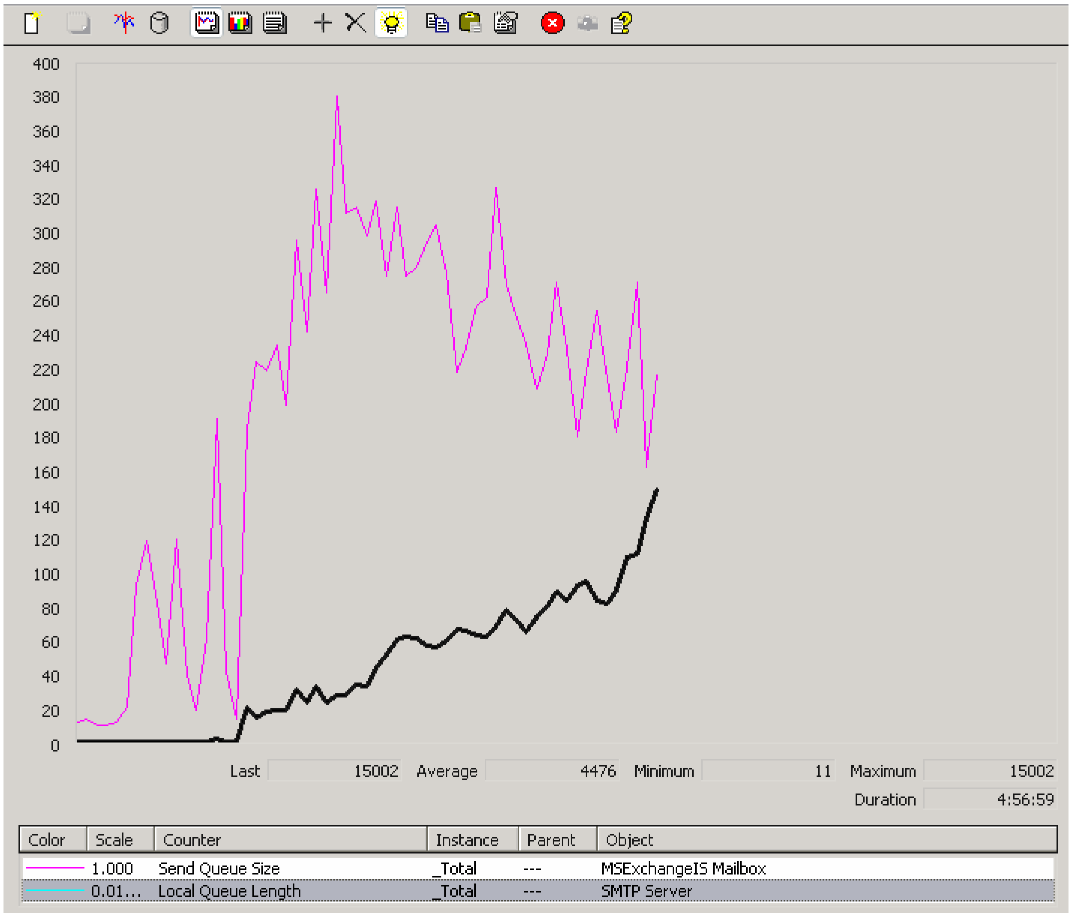


Figure 1 Monitoring mail queues using the Performance snap-in

Identifying Slow RPC Request Processing

When using Outlook in MAPI mode, clients actions in Outlook translate to remote procedure calls (RPCs) between the clients and the server. If the user is running in online mode, these RPC calls occur synchronously. Any delay by the server in fulfilling these synchronous requests directly affects user experience and the responsiveness of Outlook. Conversely, running in cached mode results in the majority of these requests being handled asynchronously. Asynchronous processing means that the speed at which most user actions are initiated should not translate into the responsiveness or overall experience of Outlook.

Generally, spikes in RPC requests that do not increase RPC operations/sec indicate that there are bottlenecks preventing the store from fulfilling the requests in a timely manner. It is relatively simple to identify where the bottlenecks are occurring with regards to RPC requests and RPC operations/sec. If the client experiences delays, but the RPC requests are zero and the RPC operations/sec are low, the performance problem is happening before Exchange processes the requests (that is, before the Microsoft Exchange Information Store service actually gets the incoming requests). All other combinations point to a problem either while Exchange processes the requests or after Exchange processes those requests.

The counters shown in Table 3 indicate delays in fulfilling RPC requests.

Table 3 Performance Counters for RPC Processing

Counter	Expectations
MSExchangeIS\RPC Requests Indicates the number of MAPI RPC requests presently being serviced by the Microsoft Exchange Information Store service. The Microsoft Exchange Information Store service can service only 100 RPC requests (the default maximum value, unless configured otherwise) simultaneously before rejecting client requests.	• It should be below 30 at all times.
MSExchangeIS\RPC Averaged Latency Indicates the RPC latency in milliseconds, averaged for the past 1024 packets.	• It should be below 50 ms at all times.

Example of Monitoring RPC Processing

In this example, the server is CPU-bound and the processors cannot handle the incoming RPC requests in a timely manner. This means that the RPC requests accumulate and have a high latency. This is shown in Figure 2 where the **MSExchangeIS\RPC Requests** counter is constantly averaging 90 and the **MSExchangeIS\RPC Averaged Latency** counter is approximately 146 milliseconds (ms).

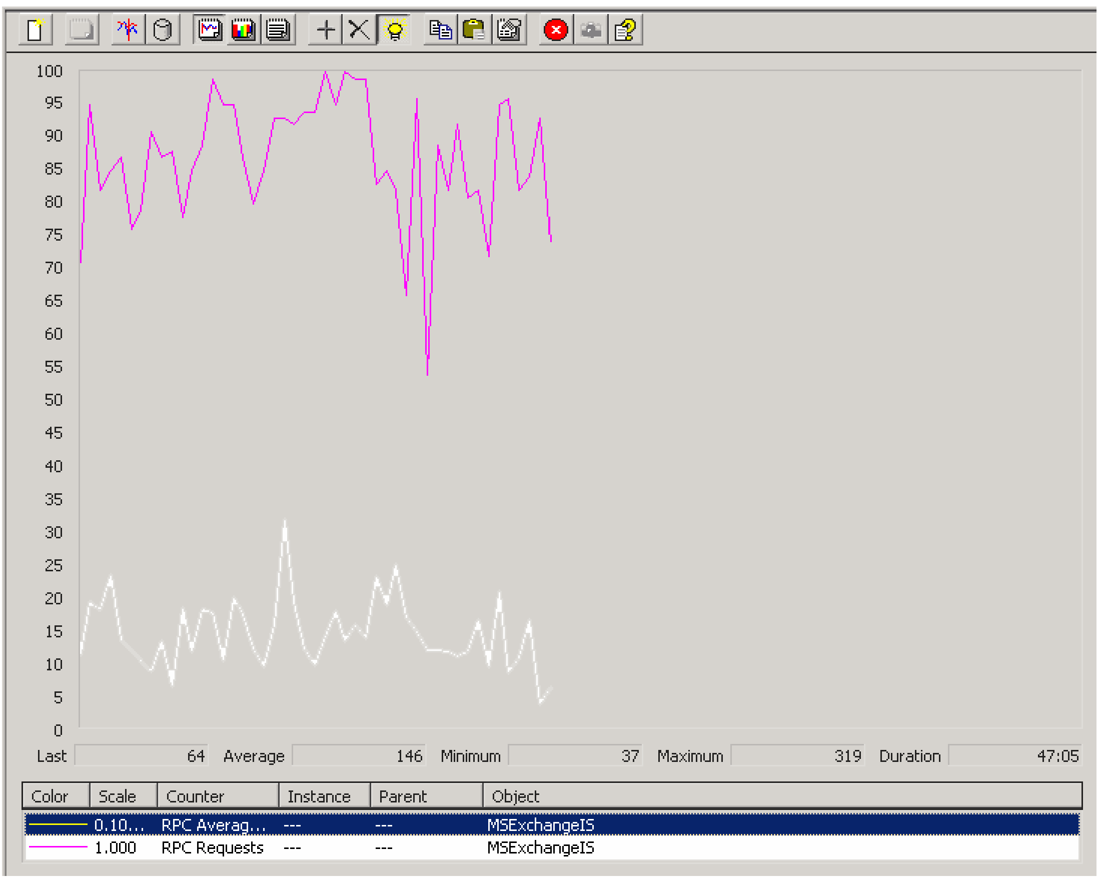


Figure 2 Monitoring RPC processing using the Performance snap-in

Identifying Rising Queues in Epoxy

Epoxy (ExIPC) is a shared memory mechanism that enables the Internet Information Services (Inetinfo.exe) and Microsoft Exchange Information Store (Store.exe) processes to quickly shuttle data back and forth. This allows for bi-directional, interprocess communication between the Inetinfo.exe process (which accepts requests from Internet protocols such as WebDAV, IMAP4, NNTP, POP3, and SMTP) and the Store.exe process. This memory is also used by DSAccess, the Exchange component that caches Active Directory information.

If there are performance issues in either the Store.exe or Inetinfo.exe processes, it is common for a queue to build up in Epoxy as one process performs faster than the other. The queues for messages sent from the Store.exe process to the Inetinfo.exe process are represented by counters called **Store Out Queue Length**. The queues for messages from the Inetinfo.exe process to the Store.exe process are represented by counters called **Client Out Queue Length**. By monitoring these counters, you can determine which queues are building up and degrading performance.

Table 4 shows the counters that you need to monitor when looking for buildups in the Epoxy queues.

Table 4 Performance Counters for Epoxy Queues

Counter	Expectations
poxy\Client Out Queue Length (WebDAV) Indicates the number of messages in the queue containing WebDAV messages sent by the Inetinfo.exe process. **Epoxy\Store Out Queue Length (WebDAV)** Indicates the number of messages in the queue containing WebDAV messages sent by the Store.exe process.	• The queues should be below 10 at all times.
Epoxy\Client Out Queue Length (DSAccess) Indicates the number of messages in the queue containing DSAccess messages sent by the Inetinfo.exe process. **Epoxy\Store Out Queue Length (DSAccess)** Indicates the number of messages in the queue containing DSAccess messages sent by the Store.exe process.	• The queues should be below 10 at all times.
Epoxy\Client Out Queue Length (IMAP) Indicates the number of messages in the queue containing IMAP4 messages sent by the Inetinfo.exe process. **Epoxy\Store Out Queue Length (IMAP)** Indicates the number of messages in the queue containing IMAP4 messages sent by the Store.exe process.	• The queues should be below 50 at all times.

Epoxy\Client Out Queue Length (NNTP) Indicates the number of messages in the queue containing NNTP messages sent by the Inetinfo.exe process. **Epoxy\Store Out Queue Length (NNTP)** Indicates the number of messages in the queue containing NNTP messages sent by the Store.exe process.	• The queues should be below 10 at all times.
Epoxy\Client Out Queue Length (POP3) Indicates the number of messages in the queue containing POP3 messages sent by the Inetinfo.exe process. **Epoxy\Store Out Queue Length (POP3)** Indicates the number of messages in the queue containing POP3 messages sent by the Store.exe process.	• The queues should be below 50 at all times.
Epoxy\Client Out Queue Length (SMTP) Indicates the number of messages in the queue containing SMTP messages sent by the Inetinfo.exe process. **Epoxy\Store Out Queue Length (SMTP)** Indicates the number of messages in the queue containing SMTP messages sent by the Store.exe process.	• The queues should be below 50 at all times unless there is a mail queue.

Example of Monitoring Epoxy Queues

In this example, POP3 users are experiencing slow responsiveness. By looking at the queue representing calls made from the Inetinfo.exe process to the Store.exe process, the Performance snap-in shows that there is a rise in the **Epoxy\Client Out Queue Length (POP3)** counter (see Figure 3). A possible cause for this rising counter is poor disk performance. In particular, there may be a disk performance issue on the Exchange store that is preventing the Store.exe process from efficiently handling incoming requests.

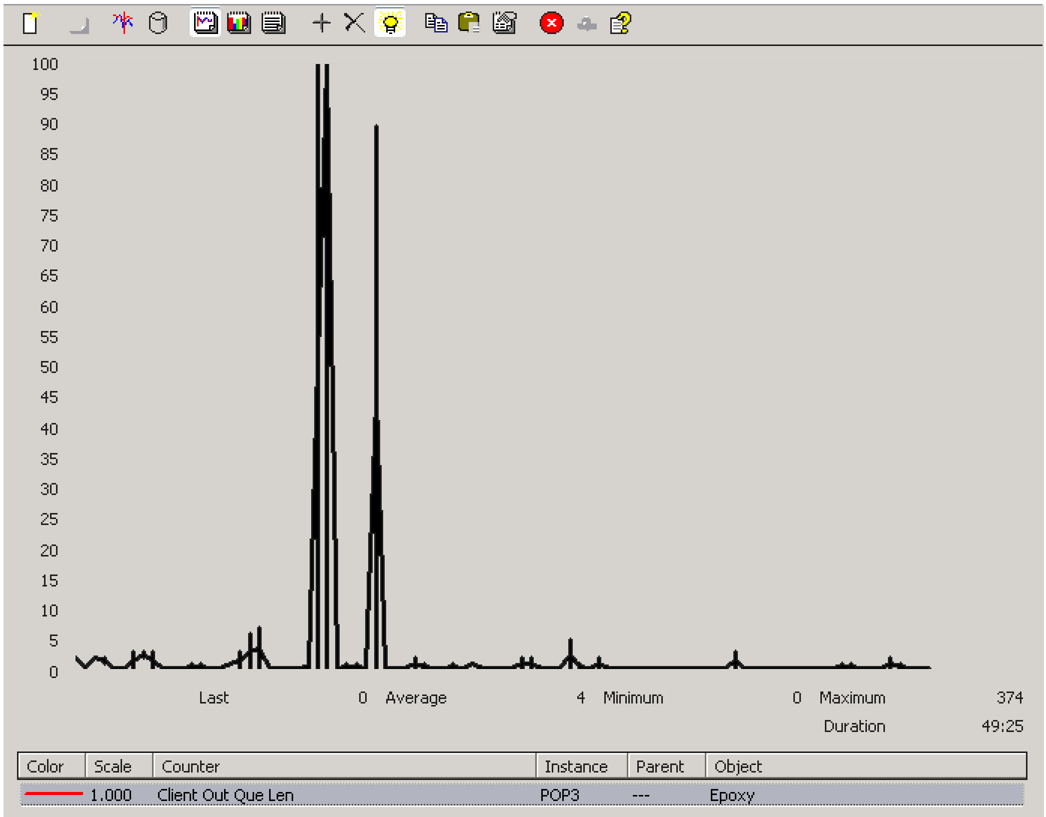


Figure 3 Monitoring an Epoxy queue using the Performance snap-in

SECTION 3

Isolating the Causes of the Degraded Performance

Having identified the symptoms of poor Exchange performance, you need to determine the cause or causes of the observed symptoms. The standard causes that negatively affect Exchange performance include:

- Third-party or support applications
- High utilization of disk space
- High utilization of memory
- Contention for processor time
- Network configuration or deployment
- Poor access to global catalogs in Active Directory

This section describes each of these causes, defines which performance counters to observe in relation to each cause, and offers suggestions on how to mitigate each cause for better performance.

Ruling Out Third-Party or Support Applications

Degraded Exchange 2003 performance can be caused by applications unrelated to Exchange, or applications that provide some support to, or build upon, Exchange. Generally, the following applications have a high impact on the performance of a server:

- Anti-virus applications
- Backup utilities
- Monitoring utilities
- Remote access tools

By using the information in this section, you can verify where the network, CPU, or I/O utilization occurs. Identifying this utilization information enables you to rule out applications that are not-Exchange related. If the problem is Exchange-related, the effect should show up in the following Exchange processes:

- Store.exe
- Inetinfo.exe
- Emsmta.exe
- Mad.exe
- Exmgmt.exe
- W3wp.exe

If any excessive I/O, CPU, or network utilization comes from processes other than these Exchange processes, check to see if any known third-party issues could be causing the problem.

The following sections explain how to understand which process is contributing to the performance degradation.

Troubleshooting I/O Transfers

If you have a disk problem, you may want to determine what is causing the I/O bottleneck. To do this, you would:

1. Identify the drive on which the I/O is occurring. If you separate the Exchange files onto separate volumes, you can more easily identify if it is the paging file, the directory database (.edb) file, the Exchange streaming database (.stm) file, the log (.log) files, or the routing queue files that are causing the I/O.

   By looking at the following counters, it is possible to understand how many I/Os are going to .edb files as compared to those going to .log files:

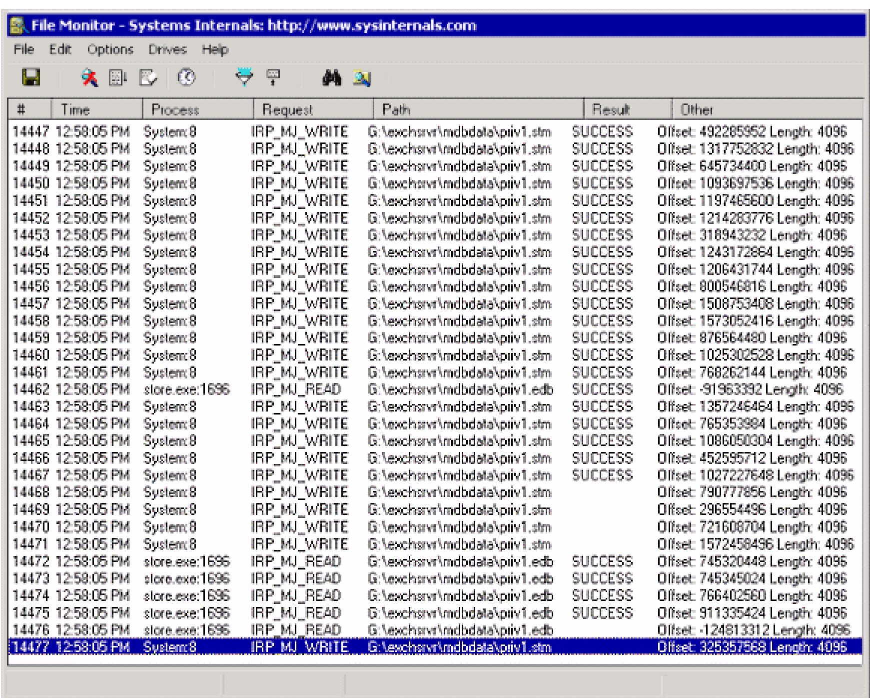
   Database\Database Pages Written/sec
   Database\Log Writes/sec

   To help determine which process is causing the disk I/O, you can use these counters:

   Process(process name)\IO Read Operations/sec
   Process(process name)\IO Write Operations/sec

2. Use the Sysinternals Filemon tool to determine which files are showing I/O activity. Choose the logical disks that need investigation and show all disk reads and writes. This is particularly useful for multi-use disks, such as C:\, which may have several major files that are used by the system or applications.

   Figure 4 shows a listing generated by Filemon for all the reads and writes to a disk.



#	Time	Process	Request	Path	Result	Other
14447	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 492285952 Length: 4096
14448	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1317752832 Length: 4096
14449	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 645734400 Length: 4096
14450	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1093697536 Length: 4096
14451	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1197465600 Length: 4096
14452	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1214283776 Length: 4096
14453	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 318943232 Length: 4096
14454	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1243172864 Length: 4096
14455	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1206431744 Length: 4096
14456	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 800546816 Length: 4096
14457	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1508753408 Length: 4096
14458	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1573052416 Length: 4096
14459	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 876564480 Length: 4096
14460	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1025302528 Length: 4096
14461	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 768262144 Length: 4096
14462	12:58:05 PM	store.exe:1696	IRP_MJ_READ	G:\exchsrvr\mdbdata\priv1.edb	SUCCESS	Offset: -91963392 Length: 4096
14463	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1357246464 Length: 4096
14464	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 765353984 Length: 4096
14465	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1086050304 Length: 4096
14466	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 452595712 Length: 4096
14467	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm	SUCCESS	Offset: 1027227648 Length: 4096
14468	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm		Offset: 790777856 Length: 4096
14469	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm		Offset: 296554496 Length: 4096
14470	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm		Offset: 721608704 Length: 4096
14471	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm		Offset: 1572458496 Length: 4096
14472	12:58:05 PM	store.exe:1696	IRP_MJ_READ	G:\exchsrvr\mdbdata\priv1.edb	SUCCESS	Offset: 745320448 Length: 4096
14473	12:58:05 PM	store.exe:1696	IRP_MJ_READ	G:\exchsrvr\mdbdata\priv1.edb	SUCCESS	Offset: 745345024 Length: 4096
14474	12:58:05 PM	store.exe:1696	IRP_MJ_READ	G:\exchsrvr\mdbdata\priv1.edb	SUCCESS	Offset: 766402560 Length: 4096
14475	12:58:05 PM	store.exe:1696	IRP_MJ_READ	G:\exchsrvr\mdbdata\priv1.edb	SUCCESS	Offset: 911335424 Length: 4096
14476	12:58:05 PM	store.exe:1696	IRP_MJ_READ	G:\exchsrvr\mdbdata\priv1.edb		Offset: -124813312 Length: 4096
14477	12:58:05 PM	System:8	IRP_MJ_WRITE	G:\exchsrvr\mdbdata\priv1.stm		Offset: 325357568 Length: 4096

Figure 4 Listing of disk read and writes produced by Filemon

Troubleshooting Processor Utilization

You should determine what is consuming the CPU. The counters below are the most likely suspects for this problem, from most likely first to least likely fourth.

Process(STORE)\% Processor Time
Process(inetinfo)\% Processor Time
Process(EMSMTA)\% Processor Time
Process(System)\% Processor Time

> **Note** Process counters count 100% for each CPU on the server. On an eight-processor computer, the value of each of the processor counters above would be between 0 percent and 800%.

Figure 5 shows a histogram view of the processes that are most likely to consume the CPU. In this figure, the Store.exe process is using up most of the CPU. If you suspect that other processes in addition to the four most likely may be hanging up the CPU, include them in this histogram view.

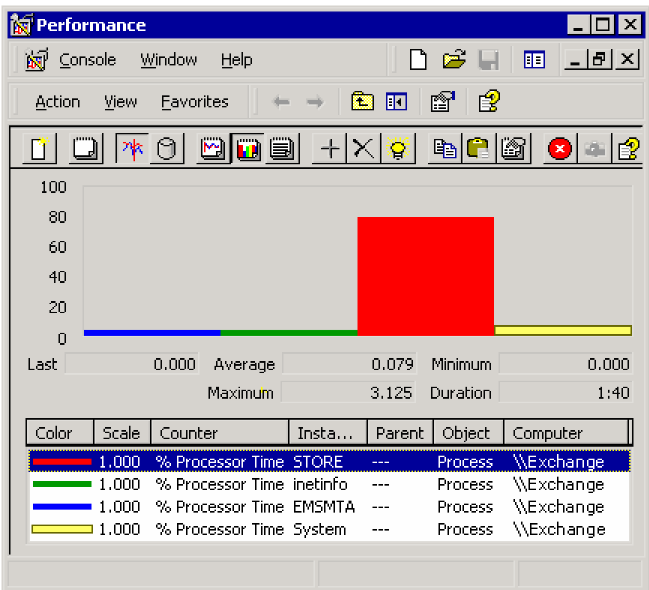


Figure 5 A histogram view of processes most likely to consume the CPU

> **Note** Looking at multiple counters in the histogram view of System Monitor (the graphing component of the Performance snap-in) is a quick way to isolate the counter indicating a problem.

Troubleshooting Memory Utilization

To determine where memory is being used, the following counters are the most likely suspects for memory consumption:

Process(process name)\Private Bytes
Database\Database Cache Size

The Store.exe process indicated by the Process(STORE.EXE)\Private Bytes counter tends to consume most of the committed bytes.

Troubleshooting Network Utilization

If client traffic is experiencing unexpected network traffic (such as heavy traffic when no clients are connected), you can use Network Monitor to examine the traffic. Network Monitor is a network diagnostic tool that monitors LANs and provides a graphical display of network statistics. While collecting information from the network's data stream, Network Monitor displays the following types of information:

- The source address of the computer that sent a frame to the network (this address is a unique hexadecimal (or base-16) number that identifies that computer on the network).
- The destination address of the computer that received the frame.
- The protocols used to send the frame.
- The data or a portion of the message being sent.

The process by which Network Monitor collects this information is called "capturing." By default, Network Monitor gathers statistics on all the frames it detects on the network into a capture buffer, which is a reserved storage area in memory. To capture statistics on only a specific subset of frames, you can single out these frames by designing a capture filter. When you have finished capturing information, you can design a display filter to specify how much of the captured information is displayed in the Frame Viewer window of Network Monitor.

To use Network Monitor, your computer must have a network card that supports promiscuous mode. If you are using Network Monitor on a remote machine, the local workstation does not need a network adapter card that supports promiscuous mode, but the remote computer does.

Once data has been captured either locally or remotely, the data can be saved to a text or a capture file, and can be opened and examined later.

> **Note** To fully troubleshoot possible network issues using Network Monitor, consider configuring Network Monitor to capture not only what the client sends and receives, but also what the server is sending and receiving. Doing both a client-side and server-side trace of network traffic helps you troubleshoot network issues more thoroughly.

Monitoring Traffic between Two Computers

To use address pairs in a capture filter, first build an address database. After building this address database, you can use the addresses listed in the database to specify address pairs in a capture filter.

To create an address database

1. In Network Monitor, on the **Capture** menu, click **Start**.

   Optionally, open a .cap file in the Frame Viewer window.
2. When you finish capturing information, on the **Capture** menu, click **Stop**.
3. On the **Capture** menu, click **View** to display the Frame Viewer window.
4. On the **Display** menu, click **Find All Names**.

   Network Monitor processes the frames and then adds them to the address database.
5. Close the Frame Viewer window, and then display the Capture window.
6. On the **Capture** menu, select **Filter**.
7. In the **Capture Filter** dialog box, double-click **Address Pairs**.

   Network Monitor displays the address database you created. You can use the names in this database to specify address pairs in the capture filter.

To monitor traffic between two computers

1. On the **Capture** menu, select **Filter**.
2. In the **Capture Filter** dialog box, double-click **ANY<->ANY**.
3. In the **Address Expression** dialog box, in the left window, select the address of one of the two computers to be monitored.
4. In the **Address Expression** dialog box, in the right window, select the address of the second computer to be monitored.
5. In **Direction**, select one of the following symbols:
   a. To monitor the traffic that passes in either direction between the two computers that you selected, select the **<-->** symbol.
   b. To monitor only the traffic that passes from the computer selected in the left window to the computer selected in the right window, select the **-->** symbol.
   c. To monitor only the traffic that passes from the computer selected in the right window to the computer selected in the left window, select the **<--** symbol.
6. Click **OK**.
7. In the **Capture Filter** dialog box, click **OK**.
8. On the **Capture** menu, click **Start**.

When troubleshooting network problems, you may need to create a capture of network traffic between two specific computers that are separated by one or more routers. In this case, you may want to analyze all network traffic between the first computer and its nearest router, and all network traffic between the second computer and its nearest router. Most of the time, this analysis determines whether network packets are being lost or corrupted somewhere between the routers. To make these traces consistent and to be able to read these traces simultaneously, the system clocks must be synchronized between the two computers before making the trace.

Additionally, it is important to understand that the bulk of Exchange network traffic is from RPC packets between the clients and the server, as well as LDAP calls to the global catalogs. Any other high network utilization that does not fit this profile should be investigated.

Ruling Out Disk-bound Problems

Disk problems are a common bottleneck for large Exchange deployments.

Exchange makes extensive use of the disk subsystem, but its use varies depending on the intended function of each disk. There are five functions that are of importance:

- Temp disk
- Database disks
- Transaction log disks
- SMTP queue
- Page file disk

Each group of disks that serve the functions above sees distinct I/O utilization patterns that require a separate analysis. Because of these different patterns, no disk should be used for more than one function.

Looking at the I/O Patterns of the Temp Disk

The operating system temporary drive is where all the format conversions, such as from RTF to HTML, occur. It is also the home for all temporary files created and accessed during crawls performed by the Microsoft Index Server Indexing Service.

When first installed, the operating system sets the location for creation and use of temporary files as the same disk used by the operating system itself. This means that any I/O for the temp disk competes with I/O for programs and page file operations being run from that drive. This competition for I/O impacts performance. To avoid having the operating system compete with for I/O with the temp disk, it is recommended that you change the global environment setting of TEMP to point to another disk and, thereby, set the temp disk to its own disk.

Use the counters listed in Table 5 to determine if there are any resource contentions in the temp disk.

Table 5 Performance Counters for Temp Disks

Counter	Expected values
PhysicalDisk\Average Disk sec/Read Indicates the average time (in seconds) to read data from the disk.	• The average value should be below 10 ms. • Spikes (maximum values) should not be higher than 50 ms.
PhysicalDisk\Average Disk sec/Write Indicates the average time (in seconds) to write data to the disk.	• The average value should be below 10 ms. • Spikes (maximum values) should not be higher than 50 ms.
PhysicalDisk\Average Disk Queue Length Indicates the average number of both read and write requests that were queued for the selected disk during the sample interval.	• The average value should be less than the number of spindles of the disk (1 if it is really a physical disk).

Looking at the I/O Patterns of Database Disks

An Exchange database consists of two files:

- **An .edb file (MAPI content)** This file stores all of the MAPI messages and tables used by the store process to locate all messages and checksums of both the .edb and .stm files, and MAPI messages.
- **An .stm file (non-MAPI content)** This file contains messages that are transmitted with their native Internet content.

Because access to either type of these files is generally random, both file types can be placed on the same disk volume.

When analyzed per physical database disk, you can use the counters listed in Table 6 to determine whether there is any performance degradation on the disks.

Table 6 Performance Counters for Database Disks

Counter	Expected values
PhysicalDisk\Average Disk sec/Read Indicates the average time (in seconds) to read data from the disk.	• The average value should be below 20 ms. • Spikes (maximum values) should not be higher than 50 ms.
PhysicalDisk\Average Disk sec/Write Indicates the average time (in seconds) to write data to the disk.	• The average value should be below 20 ms. • Spikes (maximum values) should not be higher than 50 ms.
PhysicalDisk\Average Disk Queue Length Indicates the average number of both read and write requests that were queued for the selected disk during the sample interval.	• The average should be less than the number of spindles of the disk. If a SAN is being used, ignore this counter and concentrate on the latency counters: PhysicalDisk\Average Disk sec/Read and PhysicalDisk\Average Disk sec/Write.

Example of Database Disk Monitoring

In Figure 6, one of the database disks (P:\) is experiencing high write latencies (as indicated by the **PhysicalDisk\Average Disk sec/Write** counter), averaging 62 milliseconds (ms), and frequently spiking above 80 ms and sometimes above 100 ms.

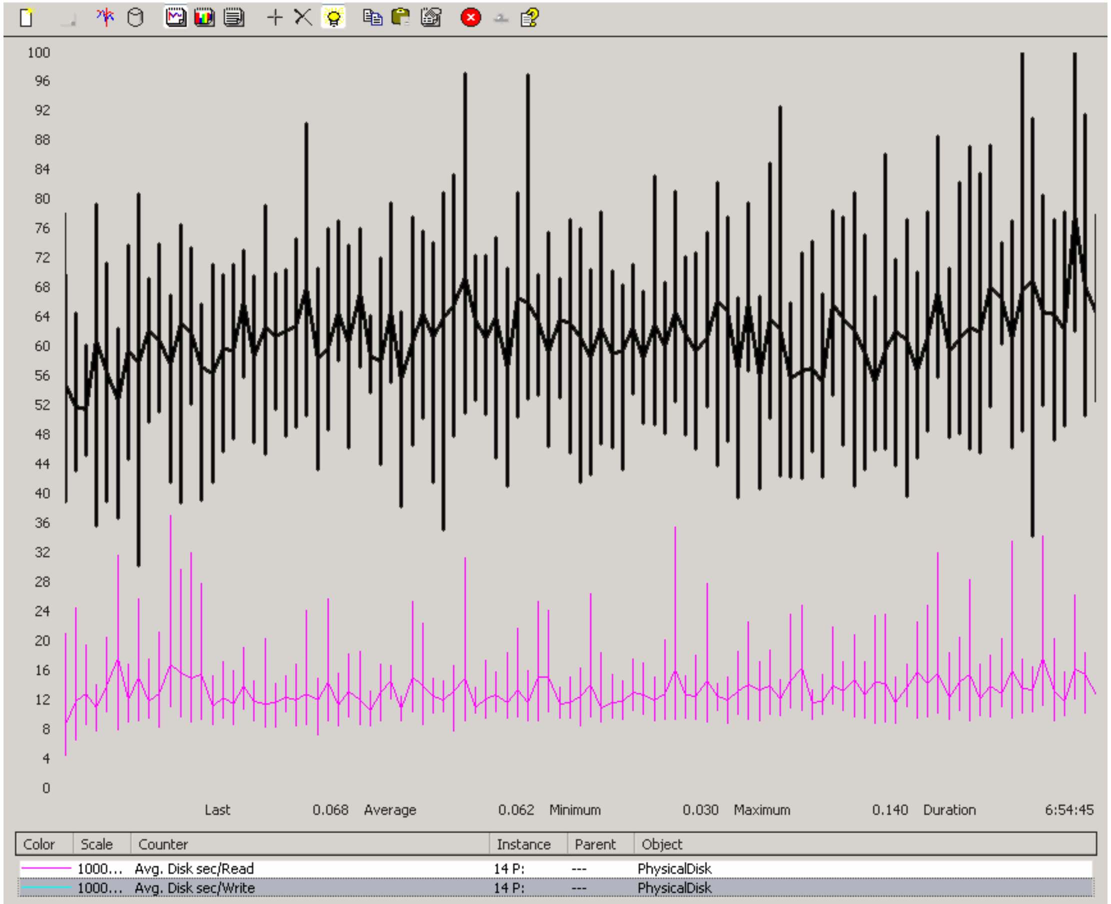


Figure 6 Monitoring a database disk using the Performance snap-in

Looking at the I/O Patterns of Transaction Log Disks

The transaction log files maintain the state and integrity of your .edb and .stm files. This means that the log files in effect, represent the data. There is a transaction log file set for each storage group. To increase performance, Exchange implements each transaction log file as a database. If a disaster occurs and you have to rebuild your server, use the latest transaction log files to rebuild your databases. If you have the log files and the latest backup, you can recover all of your data. However, if you lose your log files, the data is lost.

There are generally no reads to the log drives, except when restoring backups. This means that write performance is essential to the transaction logs and any analysis should closely observe this aspect. When analyzed per physical log disk, you can use the counters listed in Table 7 to determine whether there is any performance degradation on the disks.

Table 7 Performance Counters for Transaction Log Disks

Counter	Expected values
PhysicalDisk\Average Disk sec/Read Indicates the average time (in seconds) to read data from the disk.	• The average value should be below 5 ms. • Spikes (maximum values) should not be higher then 50 ms.
PhysicalDisk\Average Disk sec/Write Indicates the average time (in seconds) to write data to the disk.	• The average value should be below 10 ms. • Spikes (maximum values) should not be higher than 50 ms.
PhysicalDisk\Average Disk Queue Length Indicates the average number of both read and write requests that were queued for the selected disk during the sample interval.	• The average should be less than the number of spindles of the disk. If a SAN is being used, ignore this counter and concentrate on the latency counters: PhysicalDisk\Average Disk sec/Write and PhysicalDisk\Average Disk sec/Read
Database\Log Record Stalls/sec Indicates the number of log records that cannot be added to the log buffers per second because the log buffers are full.	• The average value should be below 10 per second. • Spikes (maximum values) should not be higher than 100 per second.
Database\Log Threads Waiting Indicates the number of threads waiting to complete an update of the database by writing their data to the log. If this number is too high, the log may be a bottleneck.	• The average value should be below 10 ms.

Looking at the I/O Patterns of SMTP Queues

The SMTP queue stores SMTP messages until Exchange writes them to a database (private or public), or sends them to another server or connector. SMTP queues generally experience random, small I/Os.

When analyzed per physical SMTP queue disk, you can use the counters listed in Table 8 to determine if there is any performance degradation on the disks.

Table 8 Performance Counters for SMTP Queues

Counter	Expected values
PhysicalDisk\Average Disk sec/Read Indicates the average time (in seconds) to read data from the disk.	• The average value should be below 10 ms. • Spikes (maximum values) should not be higher than 50 ms.
PhysicalDisk\Average Disk sec/Write Indicates the average time (in seconds) to write data to the disk.	• The average value should be below 10 ms. • Spikes (maximum values) should not be higher than 50 ms.

Counter	Expected values
PhysicalDisk\Average Disk Queue Length Indicates the average number of both read and write requests that were queued for the selected disk during the sample interval.	• The average should be less than the number of spindles of the disk.

Looking at the I/O Patterns of the Page File Disk

The page file serves as an extension of the physical memory, serving as an area where the system puts unused pages or pages it will need later. The page file always sees some utilization, even in machines with a good amount of free memory. This constant utilization is because the operating system tries to keep in memory only the pages that it needs and enough free space for operations. For example, a printing tool that is used only at startup might have some of its memory paged to disk and never brought back if it is never used.

In servers where the physical memory is being used heavily, it is important to ensure that all access to the page file is as fast as possible and to avoid thrashing situations. It is common for servers to start seeing errors in memory operations long before the page file is full. So, observing usage patterns of the page file disk is more important than how full the disk is. Use the counters listed in Table 9 to determine whether there is any performance degradation on the page file disk.

Table 9 Performance Counters for Page File Disks

Counter	Expected values
PhysicalDisk\Average Disk sec/Read Indicates the average time (in seconds) to read data from the disk.	• The average value should be below 10 ms at all times.
PhysicalDisk\Average Disk sec/Write Indicates the average time (in seconds) to write data to the disk.	• The average value should be below 10 ms at all times.
PhysicalDisk\Average Disk Queue Length Indicates the average number of both read and write requests that were queued for the selected disk during the sample interval.	• The average should be less than the number of spindles of the disk.
Paging File\% Usage Indicates the amount (as a percentage) of the paging file used during the sample interval. A high value indicates that you may need to increase the size of your Pagefile.sys file or add more RAM.	• This value should remain below 50%.

Improving Disk Performance

The following list describes ways to improve disk performance:

Enable caching on the array controller

For direct attach storage solutions, there is a big performance improvement when enabling the array controller's caching capabilities. In particular, the write-back cache is highly effective and should exhibit a noticeable performance improvement. It is also important to ensure that the array controller features battery backup, so any power fluctuations or outages do not cause errors or inconsistencies.

Increase log buffers

Increasing the log buffers improves the performance of a log disk that is experiencing a high number of log record stalls. For more information, see the Microsoft Exchange technical article, "Microsoft Exchange 2000 Internals: Quick Tuning Guide" (http://go.microsoft.com/fwlink/?linkid=14939) and, for Exchange 2003, start with the default value of 512 for the log buffer and increase this value in increments of 512 up to the maximum value of 9000.

Increase the database cache

Increasing the size of the database cache (dynamic buffer allocation) yields better disk performance. However, increasing the size of the database cache is not recommended because the increased size affects memory over time with the server becoming memory fragmented or running out of memory.

Align disk partitions with storage track boundaries

Aligning the disk partitions with storage track boundaries has a positive effect on disk performance. However, how significant the performance benefit is depends on the storage technology and implementation, and therefore on the storage vendor.

Enforce message size limits

Enforcing message size limits can reduce disk utilization and therefore result in better performance. Before enforcing message size limits, consider how this would affect the Service Level Agreement (SLA) of your organization.

Enforce mailbox size limits

As with enforcing message limits, enforcing mailbox quotas can result in better disk performance. Again, consider enforcing message limits, taking into consideration the SLA of your organization.

Ruling Out Memory-bound Problems

Exchange is an aggressive memory user, being able to use up to 3 GB of physical memory. On a production server, it is common to see the Store.exe process taking 1.5 GB of virtual memory, because this process maintains large memory caches.

In addition to the memory utilization by various processes within Exchange, Exchange's ExIFS kernel driver also uses kernel memory. Although less visible, high utilization of kernel memory causes severe performance degradation and instability.

Looking at User Space Memory

As the server uses memory and free memory becomes scarce, the operating system starts trimming the working set of the process and using the page file more aggressively. Using the page file affects overall performance because disk operations take longer than memory operations.

Additionally, when the paging to and from disk gets high enough, eventually a disk bottleneck occurs and performance suffers. In this case, the real problem is memory, and the disk bottleneck is only a symptom.

Use the counters listed in Table 10 to determine the current state of the user space memory.

Table 10 Performance Counters for User Space Memory

Counter	Expected values
Memory\Available Mbytes (MB) Indicates the amount of physical memory (in MB) immediately available for allocation to a process or for system use. The amount of memory available is equal to the sum of memory assigned to the standby (cached), free, and zero page lists.	• During the test, there must be 50 MB of available memory at all times.
Memory\Pages/sec Indicates the rate at which pages are read from or written to disk to resolve hard page faults. This counter is a primary indicator of the types of faults that cause system-wide delays. It includes pages retrieved to satisfy page faults in the file system cache. These pages are usually requested by applications.	• This counter should be below 1,000 at all times.

Improving User Space Memory

The following list describes how you can improve the performance of user space memory:

Remove superfluous software

To free up resources for Exchange, remove from the server any third-party software tools that do remote monitoring or any type of non-essential service. Use the Performance snap-in to understand how much memory each application consumes.

Run maintenance tasks off peak times

Running maintenance tools (such as eseutil) or tasks (such as mailbox management) during peak times can consume memory that would otherwise be needed for Exchange. It is good practice to run these tools and tasks at off peak times or during low use periods (such as weekends).

Looking at Kernel Memory Usage

The users of kernel memory include direct users (such as the kernel driver, ExIFS, in Exchange) and indirect users (such as opening and reading files). Although less noticeable than user memory issues, kernel memory is essential to the proper operation of the system and therefore you should monitor kernel memory carefully.

Use the counters listed in Table 11 to determine the current state of kernel memory.

Table 11 Performance Counters for Kernel Memory

Counter	Expected values
Memory\Pool Nonpaged Bytes Indicates the number of bytes in the kernel memory nonpaged pool. The kernel memory nonpage pool is an area of system memory (that is, physical memory used by the operating system) for kernel objects that cannot be written to disk, but must remain in physical memory as long as the objects are allocated.	• There must be no more than 100 MB of non-paged pool memory being used.
Memory\Pool Paged Bytes Indicates the number of bytes in the kernel memory paged pool. The kernel memory paged pool is an area of system memory for kernel objects that can be written to disk when they are not being used.	• Unless a backup or restore is taking place, there must be no more than 180 MB of paged pool memory being used.
Memory\Free System Page Table Entries Indicates the number of system page table entries that are available. The kernel drivers use system page table entries for holding I/O and driver data (such as video driver and network buffers) in kernel memory.	• At no point should there be less than 3,500 entries.

Improving Kernel Memory

The following list describes how you can improve the performance of kernel memory:

Minimize queues (such as SMTP delivery queues)

The presence of mail queues causes Exchange services to keep files open, which consequently has a direct effect on the availability of non-paged pool memory.

Use /VGA or a generic video driver to free up system page table entries

Video boards use the system page table entries to map their buffers in kernel space. This usage competes with Exchange's needs for system page table entries. For servers, you can free up system page table entries by using /VGA on the Boot.ini file to force booting in 800 by 600 VGA mode or by using a generic video driver.

> **Note** Changing the video resolution on the server has no effect on the video resolution of the computer from which you are running terminal services to perform remote management of the server.

Use the /Userva switch for Microsoft Windows Server™ 2003

For servers running Windows Server 2003, setting aside more memory for system page table entries helps prevent the problem of running out of system page table entries. For more information about how to increase system page table entries on the server, see Microsoft Knowledge Base article 810371, "XADM: Using the /Userva Switch on Windows 2003 Server-Based Exchange Servers" (http://go.microsoft.com/fwlink/?linkid=3052&kbid=810371).

Looking at the Exchange Store Virtual Memory

Each Store.exe process of a server has a limited amount of memory, called the store virtual memory, that it can address. As you scale a server to accommodate more users and more usage, the server may run low on virtual memory. When a server already has 4 GB of RAM, you cannot expand the memory of the server any further. Adding more physical memory cannot solve errors that indicate that you are out of virtual memory.

When a server is low on virtual memory, the server's overall performance degrades as the low memory situation forces the Store.exe process to use the page file, and the Store.exe process starts paging rapidly.

Use the counters listed in Table 12 to determine the current state of the store's virtual memory.

Table 12 Performance Counters for Exchange Store Virtual Memory

Counter	Expected values
MSExchangeIS\VM Largest Block Size Displays the size (in bytes) of the largest free block of virtual memory. This counter is a line that slopes down while virtual memory is consumed. When this counter drops below 32 MB, Exchange 2003 logs a warning (Event ID=9582) in the event log. When this counter drops below 16 MB, Exchange logs an error.	• At no point should this value go below 32 MB.
MSExchangeIS\VM Total 16 MB Free Blocks Displays the total number of free virtual memory blocks that are greater than or equal to 16 MB. This counter displays a line that may first rise, but then may eventually fall when free memory becomes more fragmented. It starts by displaying a few large blocks of virtual memory and may progress to displaying a greater number of separate, smaller blocks. When these blocks become smaller than 16 MB, the line begins to fall.	• At no point should this value go below 1.
MSExchangeIS\VM Total Free Blocks Displays the total number of free virtual memory blocks regardless of size. This counter displays a line that may first rise, but then may eventually fall when free memory first becomes fragmented into smaller blocks, and then when these blocks are consumed. Use this counter to measure the degree to which available virtual memory is being fragmented. The average block size is the Process\Virtual Bytes\STORE instance divided by MSExchangeIS\VM Total Free Blocks.	• At no point should this value go below 1.
MSExchangeIS\VM Total Large Free Block Bytes Displays the sum in bytes of all the free virtual memory blocks that are greater than or equal to 16 MB. This counter monitors store memory fragmentation and forms a line that slopes down when memory is consumed. On a healthy server, the line should stay above 50 MB.	• At no point should this value go below 50 MB.

The Store.exe process also uses its own heap allocation mechanism and structures, which are called exchmem. The Store.exe process creates several exchmem heaps at startup, and does not increase the number of heaps unless the existing number is either fully utilized or is fragmented to a point where an allocation cannot find enough contiguous memory to succeed.

If there is a memory utilization problem or internal fragmentation (fragmentation inside the exchmem heaps, which themselves reside inside the store's virtual memory space), the Store.exe process creates new exchmem heaps.

Generally, if the Store.exe process must repeatedly create additional heaps, the overall store virtual memory becomes fragmented or depleted. By tracking the counters listed in Table 13, it is possible to determine whether or not the exchmem heaps are a source of problems or performance degradation as the heaps become fragmented.

Table 13 Performance Counters for exchmem Heaps

Counter	Expected values
MSExchangeIS\Exchmem: Number of heaps with memory errors Indicates the total number of exchmem heaps that failed allocations due to insufficient available memory.	• This value should be 0 (zero) at all times.
MSExchangeIS\Exchmem: Number of memory errors Indicates the total number of exchmem allocations that could not be satisfied by available memory.	• This value should be 0 (zero) at all times.
MSExchangeIS\Exchmem: Number of Additional Heaps Indicates the number of exchmem heaps created by store after startup.	• This value should not exceed 3 at any time.

Improving Exchange Store Virtual Memory

The following list describes how you can improve the performance of Exchange store virtual memory:

Consolidate Storage Groups

For each storage group, the Store.exe process must allocate structures and consume memory. If possible, use the minimum number of storage groups that satisfy the SLA.

Offload server roles

If memory utilization increases because the server is performing multiple roles (such as being a public folder and a mailbox server), it is a good idea to offload roles to dedicated servers.

Read the Microsoft Knowledge Base Article 815372

For more information about how to optimize virtual memory usage, see Microsoft Knowledge Base article 815372, "How to Optimize Memory Usage in Exchange Server 2003" (http://go.microsoft.com/fwlink/?linkid=3052&kbid=815372).

Ruling Out Processor-bound Problems

Generally, identifying that a server is processor bound is straightforward. Use the counters listed in Table 14 to determine whether there are any contentions on the processors.

Table 14 Performance Counters for Processors

Counter	Expected values
Processor\% Processor Time (_Total) Indicates the percentage of time the processor is running non-idle threads. You can use this counter to monitor the overall utilization of the processors or per-processor.	• The average CPU utilization should be below 90% at all times.
System\Processor Queue Length Indicates the number of threads in the processor queue. There is a single queue for processor time, even on computers with multiple processors. This counter shows ready threads only, not threads that are currently running.	• This should be less than 2. • Infrequent spikes that do not correlate to the % Processor Time being above 75% are not an issue.

Example of a Processor-bound Problem

The Exchange server shown in Figure 7 is experiencing a problem due to excessive processor utilization. The CPU utilization is approximately 90% (as indicated by the **Processor\% Processor Time** counter), and the **System\Processor Queue Length** counter is consistently high.

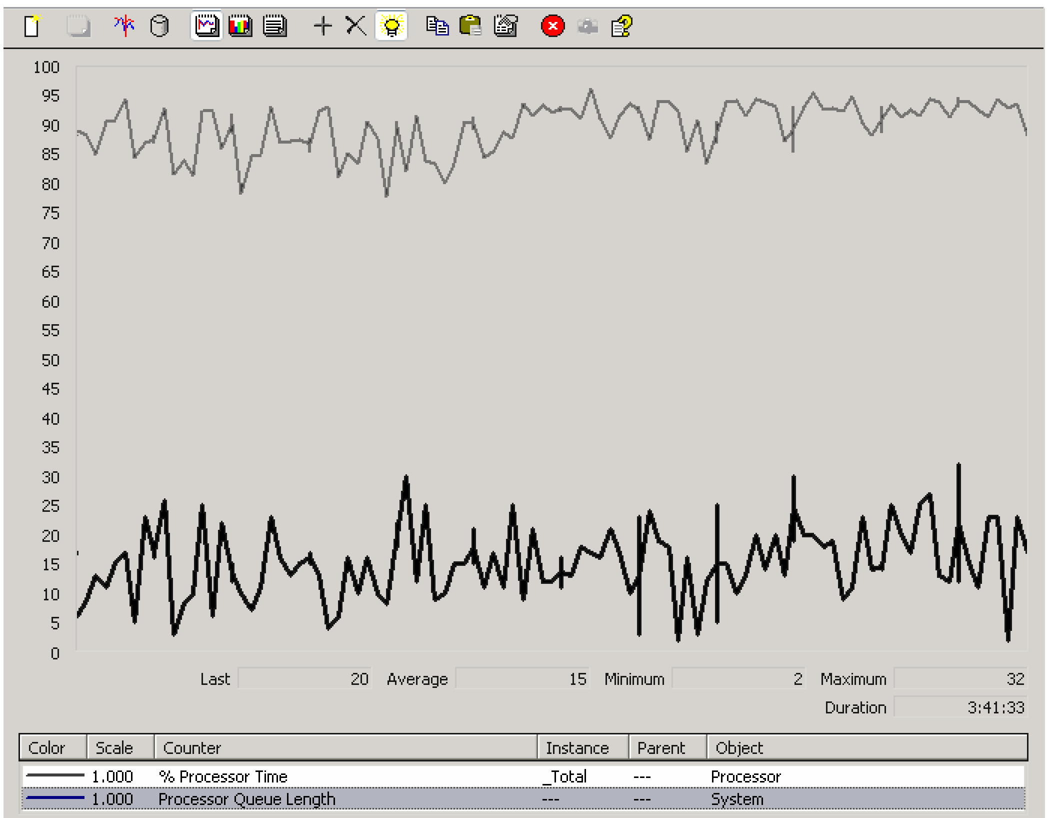


Figure 7 Monitoring a processor using the Performance snap-in

To make sure that Exchange is responsible for this CPU utilization, use the Performance snap-in to view the **Processor\% Processor Time** counter for all processes (see Figure 8). As shown in Figure 8, the Store.exe process is indeed the major cause for the CPU utilization.

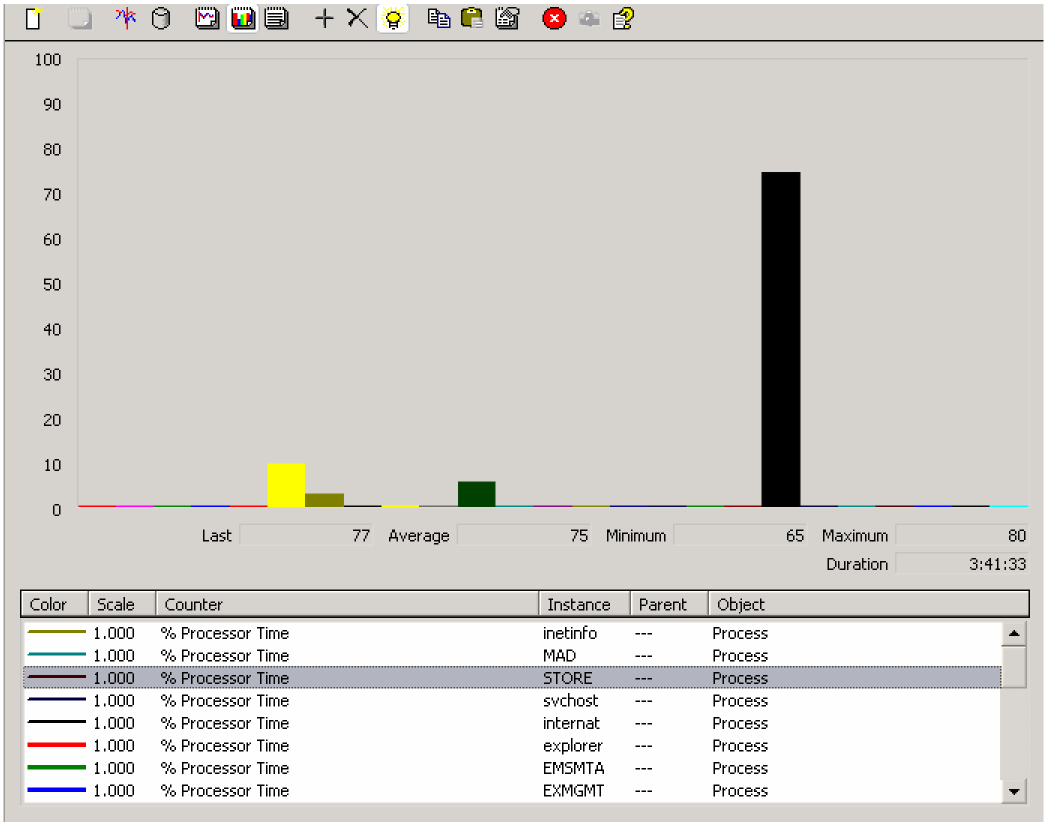


Figure 8 Monitoring the % Processor Time using the Performance snap-in

Improving Processor Performance

The following list describes how you can improve processor performance:

Ensure database maintenance and backups occur at off-peak hours and are staged

To reduce the overall impact on the server, it is important to ensure that I/O-intensive, CPU-intensive, or memory-consuming tasks (such as backup and maintenance) occur outside normal operation hours. You can further lessen the effect that these resource-intensive tasks have by staging (that is, setting different start and preferably end times for tasks) the maintenance and backup of databases or storage groups.

Offload roles to other servers

Many tasks that an Exchange server performs can be set to occur only at dedicated servers. For example, if a server sees a lot of distribution list expansions, you can reduce CPU utilization by offloading distribution list expansion to a dedicated server.

Ruling Out Network-bound Problems

The network and how it is deployed is essential to the proper performance of an Exchange server. It is uncommon for networks to be network-bound, because 100 Mbps networks generally offer enough bandwidth for most organizations. However, with increasing message sizes and users per server, it is important to ensure the network does not become a choking point.

Use the counters listed in Table 15 to determine whether there is any network performance degradation.

Table 15 Performance Counters for Network

Counter	Expected values
Network Interface\Bytes Total/sec Indicates the rate at which the network adapter is processing data bytes. This counter includes all application and file data, in addition to protocol information such as packet headers.	• For a 100-Mbps network interface card (NIC), it should be below 6–7 MB/sec. • For a 1000-Mbps NIC, it should be below 60–70 MB/sec.
Network Interface\Packets Outbound Errors Indicates the number of outbound packets that could not be transmitted because of errors.	• It should be zero (0) at all times.

Improving Network Performance

The following item describes how you can improve network performance:

Segment inter-server and global catalog traffic

When there is much traffic, and therefore overhead due to packet collision, you can improve network performance by separating inter-server and global catalog traffic from client traffic. You can do this by having servers and global catalogs with dual network adapters, and by building a separate network for the communication required by servers and global catalogs.

Ruling Out Active Directory-bound Problems

Exchange depends on the performance of the global catalog domain controllers. You can investigate CPU usage, as well as disk and memory bottlenecks, on your Active Directory servers.

> **Note** Most investigative techniques described in this article apply to global catalogs.

For each of the Exchange servers in the topology, use the counters listed in Table 16 to determine whether there is a slowdown in communicating with global catalogs.

Table 16 Performance Counters on the Exchange Server that Indicate Global Catalog Problems

Counter	Expected values
SMTP Server\Categorizer Queue Length Indicates how well SMTP is processing LDAP lookups against global catalog servers. This should be at or around zero unless the server is expanding distribution lists. When expanding distribution lists, this counter can occasionally go up higher. This is an excellent counter to tell you how healthy your global catalogs are. If you have slow global catalogs, you will see this counter go up.	• The maximum value should be below 10.
MSExchangeDSAccess Process\LDAP Read Time (for all processes) Shows the time (in ms) that an LDAP read request takes to be fulfilled.	• The average value should be below 50 ms. • Spikes (maximum values) should not be higher than 100 ms.
MSExchangeDSAccess Process\LDAP Search Time (for all processes) Shows the time (in ms) that an LDAP search request takes to be fulfilled.	• The average value should be below 50 ms. • Spikes (maximum values) should not be higher than 100 ms.

For each of the global catalogs in the topology, use the counters listed in Table 17 to determine whether the global catalogs are experiencing performance degradations.

Table 17 Performance Counters on the Global Catalog Servers that Indicate Problems

Counter	Expected values
Processor\% Processor Time (_Total) Indicates the percentage of time the processor is running non-idle threads. You can use this counter to monitor the overall utilization of the processors or per-processor.	• The average CPU utilization should be below 90% at all times.
System\Processor Queue Length Indicates the number of threads in the processor queue. There is a single queue for processor time, even on computers with multiple processors. This counter shows ready threads only, not threads that are currently running.	• This counter should be less than 2.
Network Interface\Bytes Total/sec Indicates the rate at which the network adapter is processing data bytes. This counter includes all application and file data, in addition to protocol information such as packet headers.	• For a 100-Mbps NIC, this counter should be below 6 MB/sec. • For a 1000-Mbps NIC, this counter should be below 60 MB/sec.
Network Interface\Packets Outbound Errors Indicates the number of outbound network packets that could not be transmitted because of errors.	• This counter should be zero (0) at all times.

Counter	Expected values
PhysicalDisk(NTDS Database Disk)\Average Disk sec/Read Indicates the average time (in seconds) that it takes to read data from the disk.	• The average value should be below 20 ms. • Spikes (maximum values) should not be higher than 50 ms.
PhysicalDisk(NTDS Database Disk)\Average Disk sec/Write Indicates the average time (in seconds) that it takes to write data to the disk.	• The average value should be below 20 ms. • Spikes (maximum values) should not be higher than 50 ms.
PhysicalDisk(NTDS Log Disk)\Average Disk sec/Read Indicates the average time (in seconds) that it takes to read data from the disk.	• This value should be below 10 ms at all times.
PhysicalDisk(NTDS Log Disk)\Average Disk sec/Write Indicates the average time (in seconds) that it takes to write data to the disk.	• This value should be below 10 ms at all times.
PhysicalDisk(NTDS Database or Log Disks)\Average Disk Queue Length Indicates the average number of both read and write requests that were queued for the selected disk during the sample interval.	• The average value to be less than the number of spindles of the disk. If a SAN is being used, ignore this counter and concentrate on the latency counters: PhysicalDisk\Average Disk sec/Read and PhysicalDisk\Average Disk sec/Write.
Memory\Available Mbytes (MB) Indicates the amount of physical memory (in MB) immediately available for allocation to a process or for system use. The value of this counter is equal to the sum of memory assigned to the standby (cached), free, and zero page lists.	• During the test, there must be 50 MB of memory available at all times.
Memory\Pages/sec Indicates the rate at which pages are read from or written to disk when resolving hard page faults. This counter is a primary indicator of the types of faults that cause system-wide delays. It includes pages retrieved to satisfy page faults in the file system cache. These pages are usually requested by applications.	• This counter should be below 1,000 at all times.

Improving Active Directory Performance

The following list describes how you can improve Active Directory performance:

Offload distribution list and expansion of query-based distribution groups to dedicated global catalog and Exchange servers

Expansion of distribution lists and query-based distribution groups severely affects the performance of a global catalog. You can improve performance by dedicating a global catalog for list expansions only.

Limit distribution list sizes and use nested distribution lists

To minimize the effect of performance on the global catalog, design your Active Directory deployment such that distribution lists have a limit on their size (such as 500 users), and any additional increase of distribution list members is through the use of nested distribution lists. Generally, the use of nested distribution lists yields better performance than large, single-paged distribution lists.

SECTION 4

Handling Special Exchange Roles and Performance

The previous sections listed the counters for the most common use of Exchange Servers—as mail-flow and mailbox servers. However, some organizations heavily use Exchange server roles, such as front-end servers and public folder servers. For those organizations, there are other performance issues that need to be monitored.

Looking at Front-End Servers

Front-end servers, such as those that serve Outlook Web Access, authentication, IP address checking, Secure Sockets Layer (SSL) protocol, and encryption schemes, have security features that require significant processing. For these servers, you are likely to see increased processor activity, both in privileged and user mode, and an increase in the rate of context switches and interrupts. If the processors in the server cannot handle this increased load, queues are likely to develop.

If your front-end servers are using SSL, the Isass.exe process may consume a large amount of CPU. This is because SSL processing occurs here at the server. This means that administrators used to monitoring CPU usage may see less processor consumed by the Inetinfo.exe process and more consumed by the Isass.exe process.

Improving Front-End Server Performance

The following item describes how you can improve the front-end server performance:

Use hardware cryptographic accelerators

When there is extremely high SSL use, you can improve performance by using hardware cryptographic accelerators to offload the calculations and remove SSL from being a bottleneck.

Looking at Public Folder Servers

For public folder servers, it is important to understand that the replication traffic between public folders (if there is more than one public folder in the topology) can affect all the servers involved. Arrange the replication schedule of the servers so that a replication queue does not mount any public folder. Processing replication changes causes resource competition with the operations already occurring on the server.

Use the counter listed in Table 18 to determine whether there is any public folder performance degradation.

Table 18 Performance Counters for Public Folder Server Problems

Counter	Expected values
MSExchangeIS Public\Replication Receive Queue Size Indicates the number of replication messages waiting to be processed.	• This value should not go above 100.

Improving Public Folder Server Performance

The following item describes how you can improve the public folder server performance:

Tune replication schedule to avoid queues

You can increase or decrease the frequency that a public folder replicates its content changes to other public folders. For some deployments, having replication contents replicate more frequently actually results in performance gains. These performance gains are possible because the increased replication frequency avoids big replication queues and involves less public folder content being replicated at a time.

SECTION 5

Handling Multiple Bottlenecks

It is not uncommon for servers to experience multiple bottlenecks. It is important, however, to understand whether there are any causal relations occurring—that is, where one subsystem's performance issues spills over to another subsystem. For example, a CPU-bound server can build up queues, which causes unusually high use of the SMTP disks.

Because of the possibility of causal relations occurring between subsystems, analyze the performance logs with regard to:

- The role assigned to the server.
- The cause or causes that trigger the performance degradation of one or more subsystems.

Generally, it is worth mitigating each bottleneck, and then seeing the effects of removing that malfunctioning piece of the puzzle. Otherwise, enforcing policies may be enough to mitigate issues caused by multiple bottlenecks. For instance, enforcing message sizes for POP3 retrievals can reduce the load on the database disk. However, enforcement may not be enough. There are many cases that will require upgrades or a redesign of the hardware.

Example of a Multiple Bottleneck

In this example, the Exchange server is a mailbox server that hosts 6,000 users. It is connected to three direct attach storage arrays:

- One array has the database.
- Another array has the transaction logs.
- The third array has the SMTP disks.

This Exchange server has two storage groups with two private databases, each database with 1,500 users. The SLA for backing up and restoring limits the number of storage groups to two.

The problem is that, during the daytime, users experience slow response as they use Outlook in online mode.

Looking at the Symptoms

By collecting a performance log during the eight hours of day-time use (see Figure 9) for which this server is experiencing degraded performance, it becomes clear that the **MSExchangeIS\RPC Requests** counter is constantly around 60, and that some clients experience slow responses to the operations requested. Furthermore, the **MSExchangeIS\RPC Averaged Latency** counter is constantly hitting or going above 100 ms. These are clear symptoms of performance issues that need to be isolated.

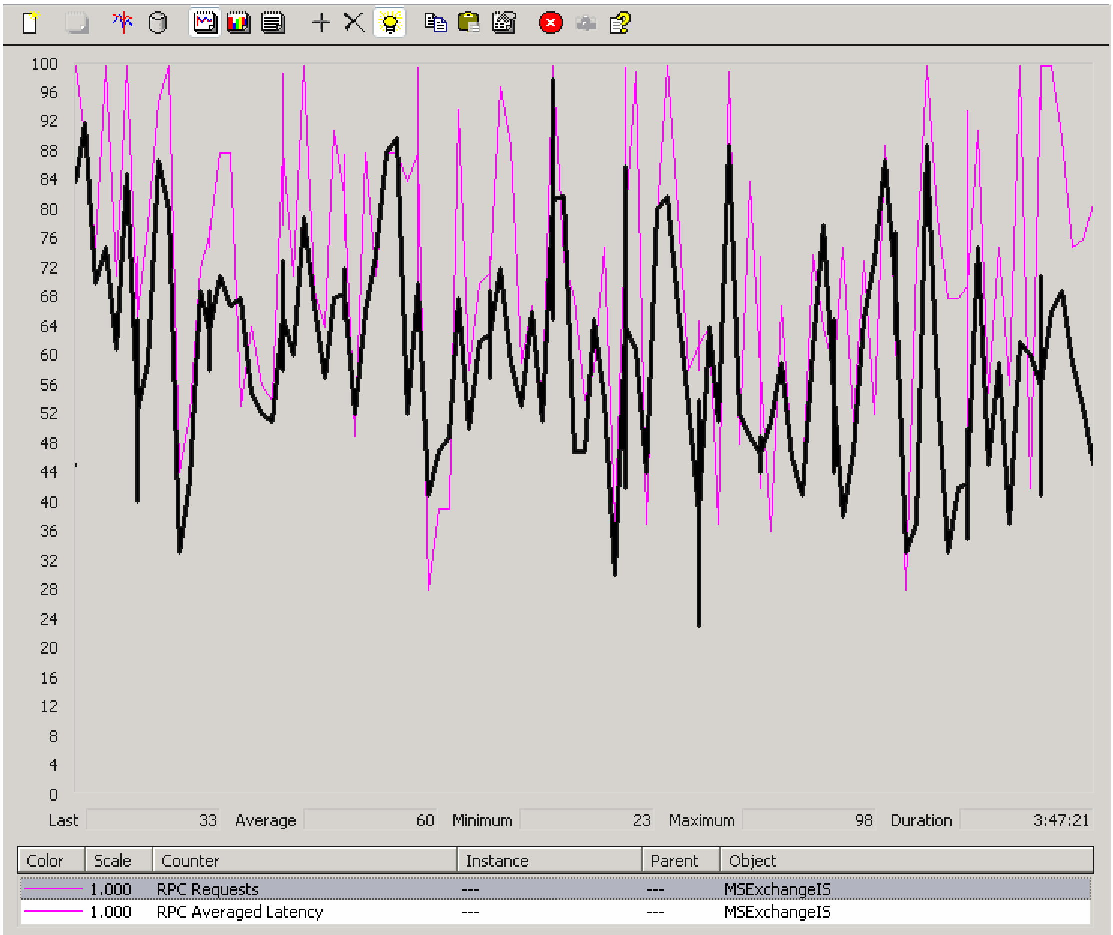


Figure 9 Eight hours of day-time performance information

Analysis of the performance logs uncovered problems with the performance of the database drive, the log drive, and the CPU. The following sections indicate which performance counters were used to determine each problem.

Problem 1—Database drive with bad performance

The Exchange Server is connected to a Storage Area Network that cannot handle the I/O load. As shown in Figure 10, the write latencies on the database drive (as indicated by the **PhysicalDisk\Average Disk sec/Write** counter) average 62 ms, with frequent spikes above 80 ms and some above 100 ms.

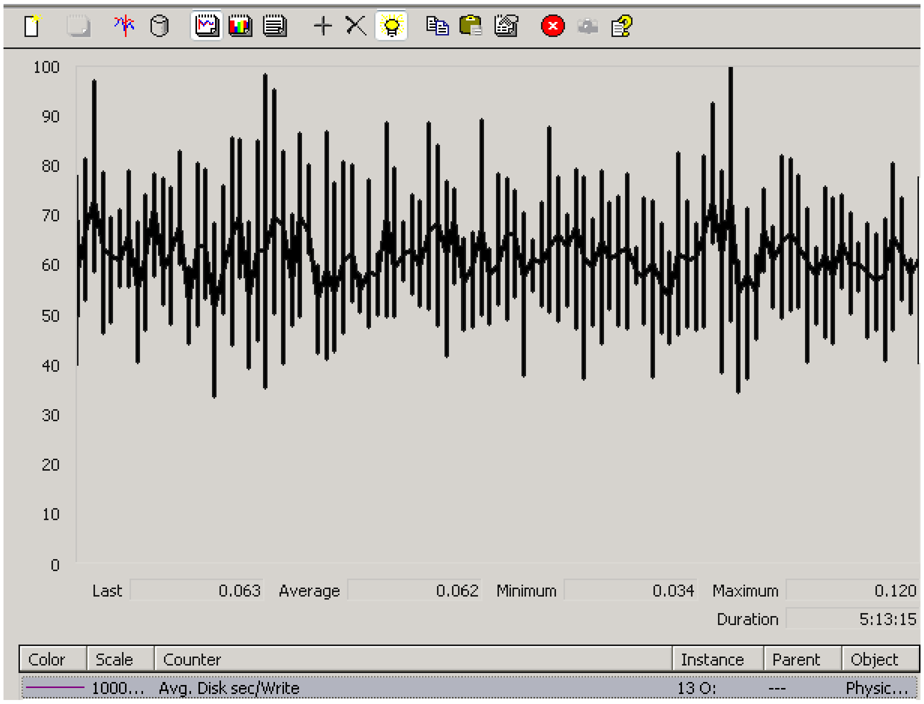


Figure 10 Performance log of the database drive

Problem 1 Solution

By adding another array and controller, and then splitting the storage groups into separate arrays, the performance of the database drive improved.

Problem 2—Log drive with bad performance

As shown in Figure 11, a slow transaction log drive is causing the **Database\Log Record Stalls/sec** counter to average 114 stalls/second, with constant spikes above 150 stalls/second. In addition, there are frequent log threads waiting as indicated by the **Database\Log Record Stalls/sec** counter, with spikes above 20.

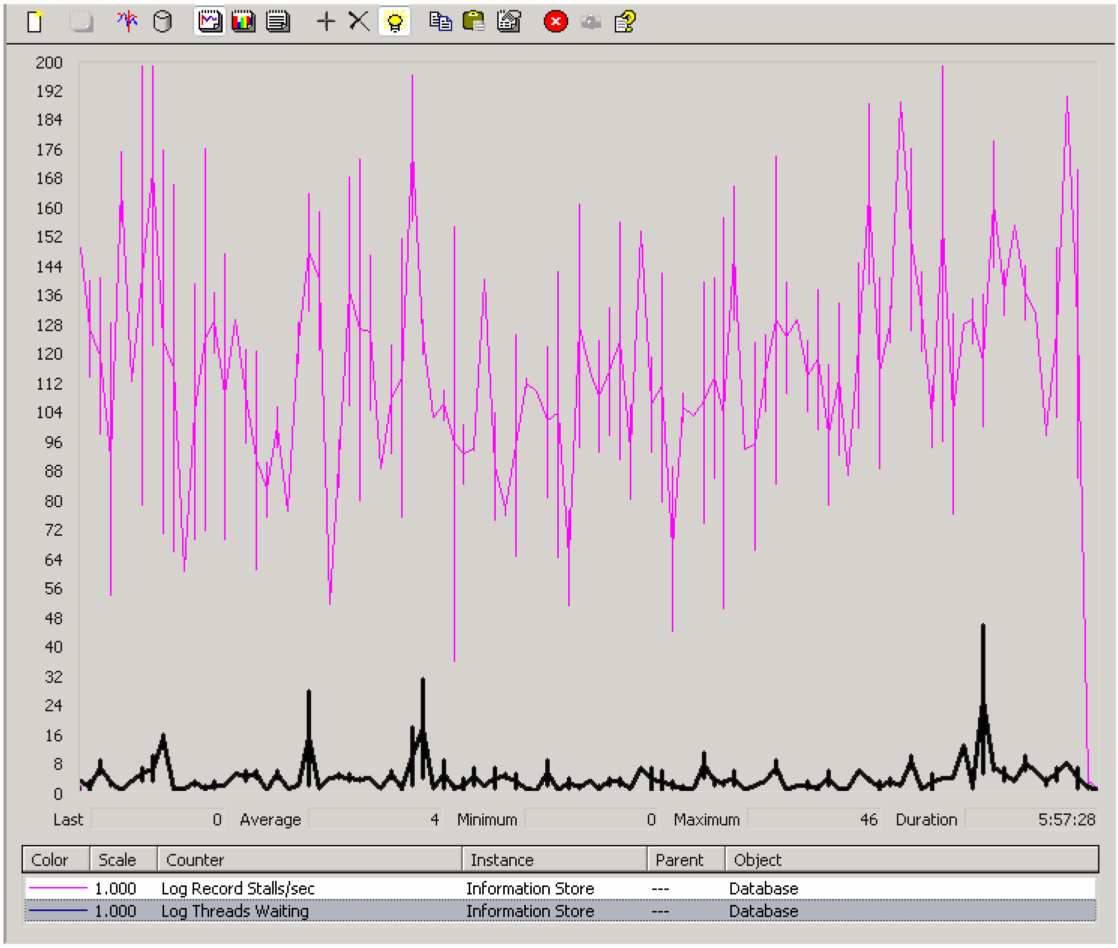


Figure 11 Performance log of the transaction log drive

Problem 2 Solution

The controller responsible for the transaction logs was experiencing problems. The controller has the write-back cache disabled. The stalls subsided after replacing the old controller with a new controller that had a properly functioning write-back cache.

Problem 3—CPU-bound

As shown in Figure 12, this server is experiencing high CPU usage (with the **Processor\% Processor Time** counter averaging 97%) and large processor queues (as indicated by the **System\Processor Queue Length** counter).

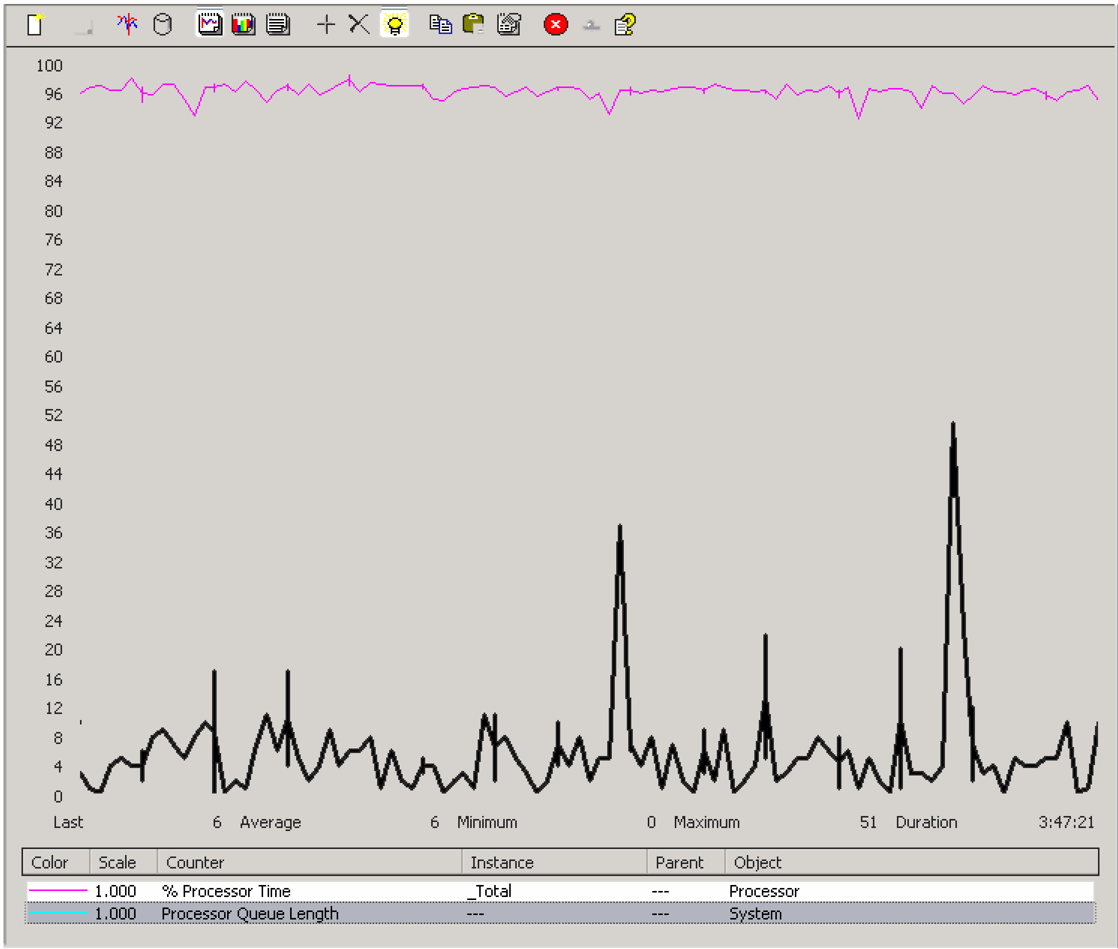


Figure 12 Performance log of CPU usage

Problem 3 Solution

The slowness on the database and transaction logs aggravated the CPU utilization, causing more context switches than necessary (an average of 50,000 on this performance log) and consequently over utilizing the CPU. By resolving the database issues in Problem 1 and the transaction logs issues in Problem 2, the CPU utilization problem shown in Figure 12 is resolved as well.

SECTION 6

Resources

The following Exchange technical article provides useful information:

- "Microsoft Exchange 2000 Internals: Quick Tuning Guide" (http://go.microsoft.com/fwlink/?linkid=14939)

The following Microsoft Knowledge Base articles provide valuable information:

- 294818, "Frequently Asked Questions About Network Monitor" (http://go.microsoft.com/fwlink/?linkid=3052&kbid=294818)
- 148942, "How to Capture Network Traffic with Network Monitor" (http://go.microsoft.com/fwlink/?linkid=3052&kbid=148942)
- 815372, "How to Optimize Memory Usage in Exchange Server 2003" (http://go.microsoft.com/fwlink/?linkid=3052&kbid=815372)
- 811237, "HOW TO: Capture Performance Data from a Remote Windows 2000 Computer using System Monitor" (http://go.microsoft.com/fwlink/?linkid=3052&kbid=811237)
- 810371, "XADM: Using the /Userva Switch on Windows 2003 Server-Based Exchange Servers" (http://go.microsoft.com/fwlink/?linkid=3052&kbid=810371)