The Complete Reference

Chapter 25

Tweaking and Optimizing Performance

827

Tweaking and optimizing performance is often a misunderstood concept so it's common for administrators to pay little attention to optimization. Maintaining a Windows Server 2003 environment is challenging enough without having to consider performance aspects. Performance optimization or tweaking is typically done when a problem such as slow logons or other performance glitches arise.

Ironically, performance optimization can significantly reduce administrators' workloads and help them get an edge over the daily grind. Combined with capacity planning, performance optimization can be used to head off problems before they become significant issues. It can assist administrators in establishing baseline metrics on which to gauge how the network or system operates. Future performance measurements can be compared to those baselines to determine how to streamline performance. Performance optimization can also be used to handle tasks, such as how to size a system, when to upgrade, and when to segment workloads.

No matter the circumstances, you can see how performance optimization and capacity planning affect you, your environment, and users' perceptions. So how can you change users' perceptions? How can you create or maintain a reliable, efficient Windows Server 2003 environment and minimize or eliminate fire fighting? These questions and many more are addressed in this chapter. By no means will this chapter be your savior for all computing problems, but it will give you a solid understanding of why you need to adhere to performance optimization and capacity planning procedures. The beauty of these procedures is that they can be applied to small environments and scale well into enterprise-level systems.

Examining Performance Optimization

Performance optimization has matured over the years for Windows-based environments into a way to make systems run faster, handle bigger workloads, and ensure reliability, availability, and serviceability of computer resources. Engineers and administrators responsible for systems can appreciate the need for proactive monitoring to provide adequate support to end users and the business structure.

With the increasing popularity of Windows systems in the business world, the responsibilities now placed on Windows Server 2003 are far greater than those placed on its predecessors, Windows NT and Windows 2000. As a result, performance optimization is crucial to the successful management of your environment.

Performance optimization coupled with capacity planning is one of the most important and most difficult responsibilities you face with both small- and large-scale Windows Server 2003 environments. It requires a combination of disciplines and can always be improved upon because work habits and environments continually change. Performance optimization encompasses many aspects of systems management, performance management, deductive reasoning, and forecasting. However, there is more to performance optimization than just using formulas or statistical information. You must use your subjective, creative, and intuitive insight in addition to relying on purely analytical solutions.

When faced with performance optimization aspects, it's important to apply the focus of the business along with technical principles so that the organization as a whole benefits. Just because a system can perform operations blazingly fast doesn't necessarily mean that the organization is adequately serviced or that resources are properly used. Some key questions to keep in mind while tweaking or optimizing performance are the following:

- How quickly can a task be accomplished?
- How much work can be performed?
- What costs are associated with different business strategies?
- Is the system used effectively and efficiently?

Performance optimization combined with capacity planning enables you to stay one step ahead of your system and anticipate future resource requirements by evaluating existing system behavior. It also helps define the overall system by establishing baseline performance values and then, through trend and pattern analysis, providing valuable insight into where the system is heading. It is an invaluable aid for uncovering both current and potential bottlenecks. Properly implemented performance optimization procedures can reveal how specific system management activities (software and hardware upgrades, changes in network topologies, and so on) may affect performance, future resource requirements, and budgeting strategies. Performance optimization allows you to attend to performance issues proactively instead of retroactively.

Establishing Service Levels and Goals

Performance optimization seeks to balance resources and workloads. It is extremely difficult to provide just the right amount of computing power for the tasks to be performed. If a system is powerful but underutilized, then a lot of resources are of little value and a waste of money. On the other hand, if a system cannot handle the workload, then tasks or transactions are delayed, opportunities are lost, costs increase, and the user (or customer) perceives a problem. Thus, a primary goal of performance optimization is *balance.*

Performance optimization involves working with unknown or immeasurable aspects of a system, such as the number of transactions the system will need to perform in the next few months or years. Other issues may relate to administration workload capacity, such as the number of system administrators that will be needed to maintain the operability of the company's database server. All of these questions are related to performance optimization and capacity planning methodologies, and their answers cannot be predicted with complete accuracy. Estimating future resource requirements is not an easy task. However, performance optimization provides a process in which you can establish benchmarks and analyze characteristics of present system resource utilization and use these to make predictions about future needs. Your level of understanding and control of your system needs is limited; to achieve a balance between capacity and workload, you must gain as much understanding and control of the environment as

possible. Controlling the aspects that are within your reach greatly increases your chances of successfully maintaining the reliability, serviceability, and availability of your system.

To begin proactively managing your system, it is important to establish *system-wide policies and procedures.* Policies and procedures help define service levels and shape users' expectations. Once these are defined, you can easily begin characterizing workloads, which will, in turn, help you define the *baseline performance values* needed to gauge the health of your system. For instance, you can define a service level agreement that states that response times for a particular system will be three seconds or less.

Establishing Policies and Procedures

The policies and procedures you decide to implement depend entirely on your network environment. The process of defining levels of service and objectives for your system gives you a certain level of control over the system's resources. Without this level of control, it is difficult to understand a system's intricacies much less manage and optimize system performance. Policies and procedures also help you winnow out empirical data and transform it into information that you can use to determine current as well as future capacity requirements. In essence, policies and procedures define how the system is supposed to be used, establishing guidelines to help users understand that they can't always have total freedom to use system resources any way they see fit. In a system where policies and procedures are working successfully and where network throughput suddenly slows to a crawl, you can assume that the reason is not, for instance, that some people were playing a multiuser network game or that a few individuals were sending enormous e-mail attachments to everyone throughout the company.

Consider establishing two sets of policies and procedures: one set that you communicate to users, and one set that the information systems (IS) department and systems support staff use internally. For example, policies and procedures for users might include a limitation on the size of e-mail attachments and discouragement of the use of beta products (other than ones internally developed) on your network. Internal policies or procedures might include rules that all backups should be completed by 5 A.M. each workday and that routine system maintenance (server refreshes, driver updates, and so on) should be performed on Saturday mornings between 6 and 9 A.M. The following list provides additional examples of policies and procedures that might be applied to your environment:

- Specify that computing resources are intended for business use only—that is, that no gaming or personal use of computers is allowed.
- Specify that only certain applications are supported and allowed on the network.
- Establish space quotas on private home directories while enforcing these policies through quota management software provided within Windows Server 2003.
- Establish replication intervals for certain databases.
- Specify that users must follow a set of steps to receive technical support.

Note *It's important to understand what users expect from the system. This can be determined through interviews, questionnaires, and the like.*

Establishing Baseline Values

By now you may be asking, "What do I do to begin performance monitoring?" or "How do I size a new Windows Server 2003 network or server?" In fact, you've already begun the process by defining policies and procedures, which cut down the amount of empirical data that you face. The next preparatory step for performance optimization is establishing baseline values so you can monitor performance. You need a starting point to which you can compare results. In determining baseline values, you deal with a lot of hard facts (statistical representations of system performance), but there are also a few variables that require your judgment and intuition. These variables are workload characterization, benchmarks, vendor-supplied information, and of course, your data collection results. Later on you can compare the baseline with the current metrics to troubleshoot, perform trend analyses, and more.

Workload Characterization

Identifying the *workloads* of a system can be an extremely challenging task, in part because resources often intertwine among different workloads and vary in processing time as well as in the amount of data being processed. Workloads are grouped, or characterized, according to the type of work being performed and the resources used. The following list shows how workloads can be characterized:

- Department function (research and development, manufacturing, and so on)
- Volume of work performed
- Batch processing
- Real-time processing
- Service requests needing attention within a specified time
- Online transactions

Once you have identified your system's workloads, you can determine the resource requirements for each and plan accordingly. This process will also help you understand the performance levels the workloads expect and demand from the system. For example, some workloads may be more memory intensive than processor intensive.

Benchmarks and Vendor-supplied Information

Benchmarks are values that are used to measure the performance of products such as processors, video cards, hard disk drives, applications, and entire systems. They are among the most sought-after performance indicators in the computer industry. Almost

every company in the computer industry uses these values to compare itself to the competition. As you might suspect, benchmarks are used heavily in sales and marketing, but their real purpose is to indicate the levels of performance you can expect when using the product.

Most benchmarks are provided by the vendors themselves, but they can originate from other sources, such as magazines, benchmark organizations, and in-house testing labs. Table 2-1 lists organizations that provide benchmark statistics and tools for evaluating product performance. Benchmarks can be of great value in your decision-making process but they should not be your only source for evaluating and measuring performance. When consulting benchmark results during capacity planning, use them as guidelines only and use care in their interpretation.

Data Collection: What Is Being Monitored

Each Windows Server 2003 system has components that the Performance snap-in can monitor. These components can be hardware or software components that perform tasks or support workloads. Many of these components have indicators that reflect certain aspects of their functionality that can be accurately measured in terms of the rate at which tasks are accomplished. For example, the Network Segment: Total bytes received/second counter shows you the number of bytes placed on the Windows Server 2003 system by the network subsystem. All collected data comes from the counters that the Performance snap-in monitors.

Objects

In Windows Server 2003 systems, many of the components that comprise an entire system are grouped into *objects* based on their characteristics. For example, anything pertaining to the processor is located in the Processor object, and anything pertaining to memory is located in the Memory object. Objects are grouped according to functionality or association within the system. They can represent logical mechanisms, such as processes, or physical entities, such as hard disk drives.

The number of objects isn't limited to what Windows Server 2003 provides. All Microsoft BackOffice products have objects that can be evaluated and tracked by the Performance snap-in or similar performance monitoring tools. Objects can also be created by third-party vendors, so that you, as an IT professional, can use a tool like

Organization Name	Web Address
Transaction Processing Performance Council	www.tpc.org
Computer Measurement Group	www.cmg.org

Table 25-1. *Organizations That Provide Benchmarks*

the Performance snap-in to monitor your own components. Microsoft has purposely chosen to let outside vendors create objects and counters specific to their own applications or devices that these tools can read.

The number of objects present on a system depends on the system configuration. For instance, Internet Information Server counters won't be present if the system isn't running that application. However, a few of the common objects that can be found in every system are the following:

- Cache
- Logical disk
- Memory
- Paging file
- Physical disk
- Process
- Processor
- Server
- System
- Thread

Counters

Each object contains *counters.* Counters typically provide information about use, throughput, queue length, and so on for a particular object. For example, all counters pertaining to the paging file are contained in the Paging File object. Performance optimization tools use the counters within an object to collect data. The information gathered from these counters is then displayed in the tool's window or dumped into a data file.

Instances

If your system has more than one similar component (two hard drives, four processors, and so on), each one is considered an instance of that component. Each instance in the system has an associated counter that measures its individual performance. Counters with multiple instances also have an instance for the combined instances.

Performance Monitoring Tools

A growing number of tools are available for collecting and analyzing system data and forecasting system capacity on the Windows Server 2003 platform. Microsoft offers some useful utilities that are either built into Windows Server 2003 or sold as separate products that can be used to collect and analyze data. These include Task Manager,

Network Monitor, and Performance snap-in (also known as Performance Monitor), which are built into the operating system, as well as Microsoft Operations Manager (MOM) and Systems Management Server (SMS), which are stand-alone products. Data collected from these applications can be exported to other applications, such as Microsoft Excel or Access, for storage and analysis.

Task Manager

The Windows Server 2003 Task Manager provides multifaceted functionality. It allows you to monitor system activity in real time and to view processor, memory, application, and process status information. You can switch to other running applications or processes, and you can easily end a task or process.

To start using Task Manager, you can use any of the following three methods:

- Right-click the taskbar and select Task Manager.
- Press CTRL-SHIFT-ESC.
- Press CTRL-ALT-DELETE and then click Task Manager.

When you execute Task Manager, the screen that you see in Figure 25-1 will appear.

Figure 25-1. *The Task Manager window after initial startup*

This window contains five tabs—Applications, Processes, Performance, Networking, and Users—that you can toggle among. In addition, a status bar at the bottom of the window displays the number of running processes and the percentage of CPU and memory used.

Task Manager presents valuable real-time performance information that can help you determine what processes or applications are problematic and give you an overall picture of the health of your system. Unfortunately, its limitations, such as its inability to store collected performance information and the breadth of its monitoring capabilities, do not make it a prime candidate for performance optimization purposes. Moreover, it can give you information pertaining to the local machine only. You must be physically at the machine to gauge performance with Task Manager.

Network Monitor

There are two flavors of Network Monitor that can be used to check network performance. One is packaged within Windows Server 2003, and the other is a component of SMS. Both versions have the same interface, as shown in Figure 25-2, and many functional components, but there are a few differences in what they can monitor.

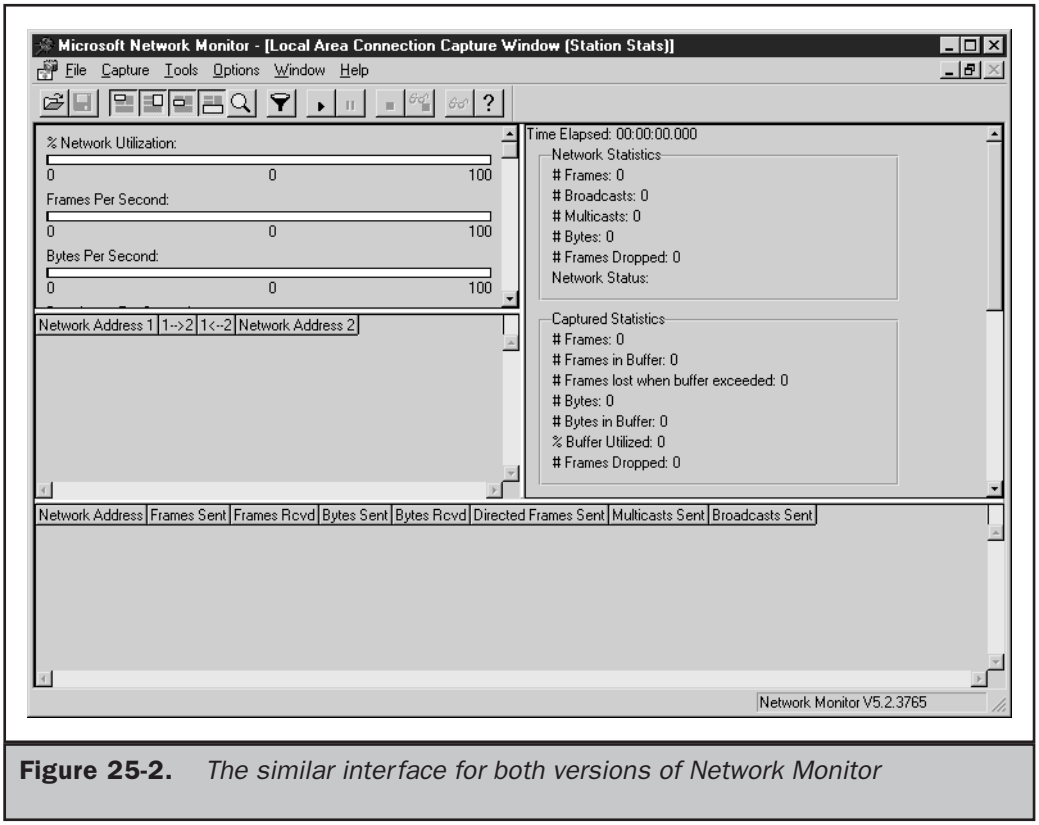


Figure 25-2. *The similar interface for both versions of Network Monitor*

Network Monitor, built into Windows Server 2003, is intended to monitor only the network activity on the local machine. For security reasons, you cannot capture traffic on remote machines. Network Monitor can, however, capture all frame types traveling into or away from the local machine.

To install Network Monitor, do the following:

1. Choose Start | All Programs | Control Panel | Add or Remove Programs.
2. Select Add or Remove Windows Components.
3. Highlight Management and Monitoring Tools and then click Details.
4. Select Network Monitor Tools and then click OK.
5. Click Next and then click Finish when the installation is complete.

To use Network Monitor, simply select it from the Start | Administrative Tools menu.

The SMS version of Network Monitor is essentially an enhanced version of the one integrated into Windows Server 2003. The primary difference between them is that the SMS version can run promiscuously throughout the network and monitor remote machines. In addition to monitoring remote machines, it can find routers present on the network, monitor the traffic circulating through the network, and resolve addresses from names.

Caution *The SMS version of Network Monitor presents possible security risks because of the nature of its monitoring techniques and privileges. It can monitor network traffic traveling into and away from remote machines. Any sensitive data that Network Monitor captures could possibly be revealed. Consequently, it is imperative that you limit the number of administrators or IS staff members who can use this version of Network Monitor.*

The SMS version of Network Monitor coincides more with performance optimization objectives because it can monitor several machines at once from a centralized location. Using the Windows Server 2003 version limits the scope of your monitoring and data collection. It also forces you to install management and monitoring tools on every machine that needs to be monitored. This results in additional memory requirements and processing power for each machine. For performance optimization and capacity planning purposes, the SMS version of Network Monitor is an excellent tool for providing real-time network analysis and establishing historical network performance statistics that can be used to examine the health of your network.

Performance Snap-in

The Performance Microsoft Management Console (MMC) snap-in is the most commonly used performance monitoring tool, both because it is bundled with the operating system and because it allows you to monitor every system object that has measurable counters

associated with it. The Performance snap-in has two tools: System Monitor and Performance Logs and Alerts. The Performance snap-in is located within the Administrative Tools group on the Start menu. Figure 25-3 shows the System Monitor startup screen.

The Performance snap-in is an excellent tool because it allows you to analyze data through charts, reports, and logs that you can save for future scrutiny. This chapter assumes that you will use the Performance snap-in as your performance optimization tool since it is available to everyone running Windows Server 2003 and its principles can be applied to other utilities.

Charting Performance with System Monitor

Counter statistics can be monitored in real time with System Monitor. The results of the collected data appear in a histogram (bar chart) or graph. The graph format is the default format for System Monitor and it produces a chart that looks something like an electrocardiogram used for monitoring a heartbeat. The charting format you choose is determined mainly by personal preference. You may find one format more suitable than others for viewing your system.

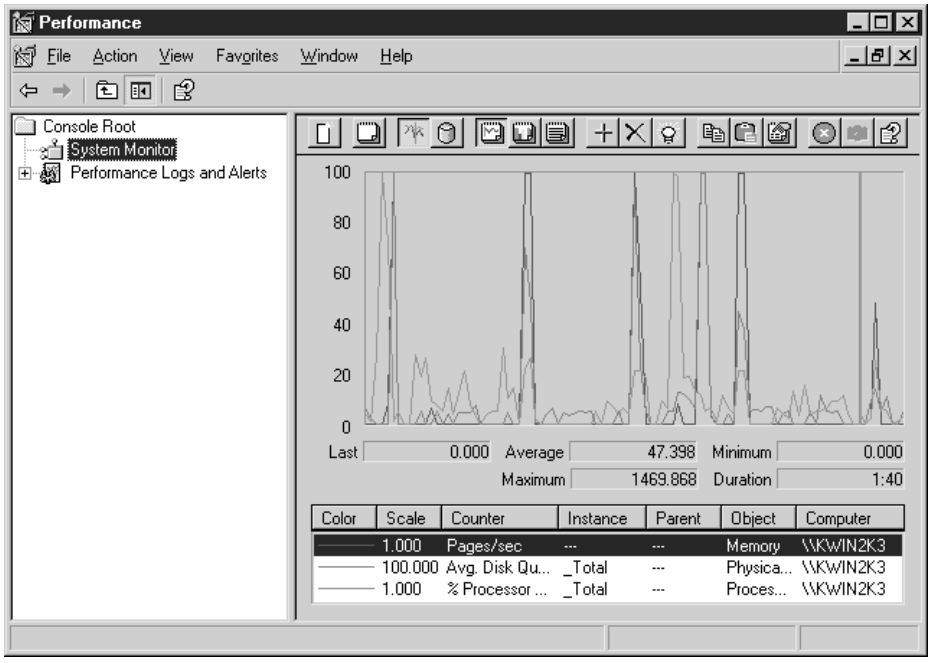
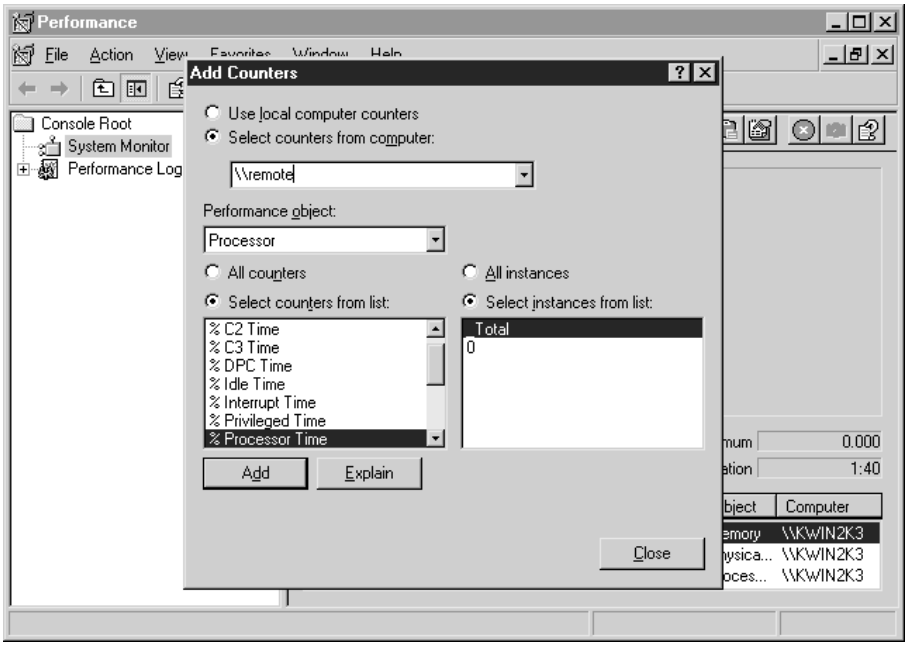


Figure 25-3. *The Performance snap-in startup screen*

To add counters to view in real time after opening the Performance snap-in from the Start | Administrative Tools menu, follow these steps:

1. Ensure that the System Monitor in the left pane is selected and then click the + button in the right pane to add counters to monitor.
2. If you want to monitor a remote machine, enter the UNC name of the computer in the Add Counters window.



3. Choose the object you want to monitor.
4. Select the desired counters within the object; you can also select the All Counters option to select all the counters for a given object.

Note *If you're not sure whether to add a certain counter, click Explain to gain a better understanding of the particular counter.*

5. Click Add to add the counter to your monitoring scheme.
6. Add more counters if desired.
7. Click Close when you are finished.

Note *You can highlight an individual counter by selecting the counter and pressing CTRL-H. There's also an easier way now with System Monitor: just select the counter you want to highlight and click the highlight button in the right pane. This helps you differentiate that counter from the rest.*

Toggling Views The default view for System Monitor is the graph format. However, you can easily switch to histogram or report viewing. Using the button bar, you can choose among any of these formats. For instance, you can click the View Report button to view the real-time data in a report format, as shown in Figure 25-4.

Performance Logs and Alerts

Working in conjunction with System Monitor is the Performance Logs and Alerts service. It stores the data it collects in a data file, or log. Logged data isn't viewed in real time, so logging provides a historical perspective on system performance. Logging is the preferred approach in performance optimization because it makes it easier to interpret trends or patterns in system performance. It also provides a mechanism for storing data in a convenient format for future scrutiny. You can use System Monitor to replay the cataloged performance data, or you can easily export it to other applications.

Note *Because Performance Logs and Alerts runs as a service, you don't have to be logged on to collect vital system statistics.*

Performance Logs and Alerts serves three functions: it monitors counters, collects event traces, and provides an alerting mechanism. These three functions can be found by expanding Performance Logs and Alerts in the Performance snap-in.

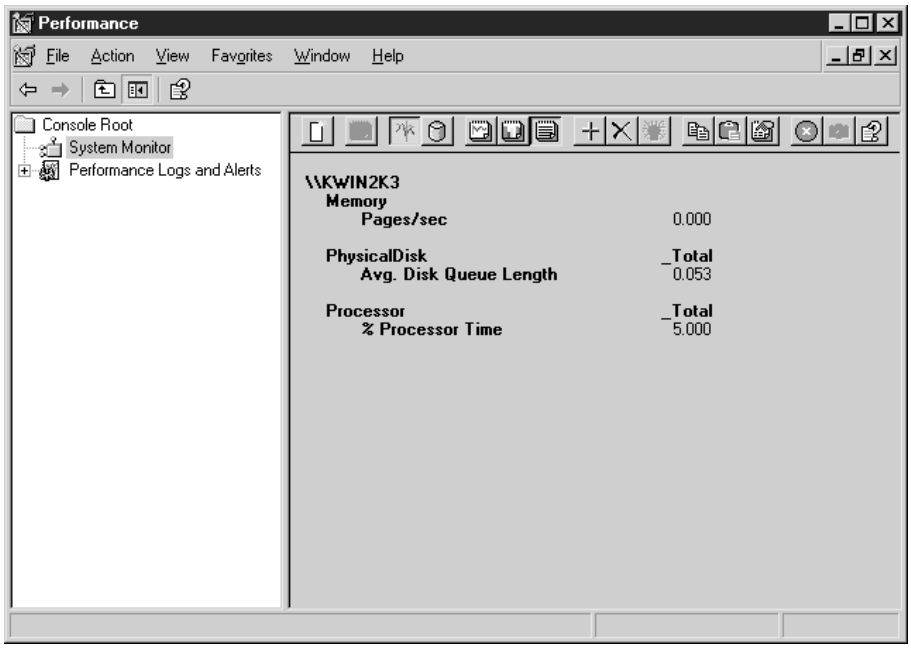


Figure 25-4. *Viewing real-time data with the report format*

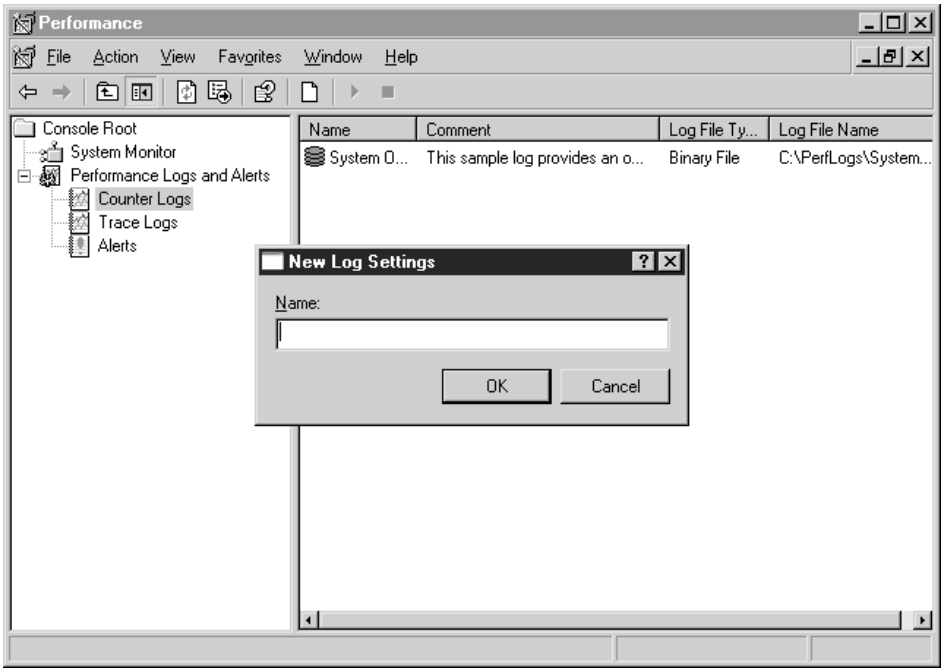
Working with Counter Logs

Counter logs allow you to record system activity or usage statistics for local and remote machines. In addition to starting and stopping the Performance Logs and Alerts service manually, you can also configure the service to start and stop automatically or to log data continuously.

Note *You can log data from individual counters or entire objects. This provides the flexibility to keep the amount of data you're logging to a minimum.*

To begin logging activity using the counter logs, follow these steps:

1. Start the Performance snap-in by selecting Performance from the Start | Administrative Tools menu.
2. Expand Performance Logs and Alerts and then select Counter Logs.
3. Choose New Log Settings from the Action menu, or right-click in the right pane and select New Log Settings. You'll be asked to supply a name for the log.



4. After you name the log, click OK; a properties window for the new log file appears, as shown in Figure 25-5. On the General tab, click Add Counters or Add Objects depending upon what you want to monitor.

5. In the Add Objects or Add Counters window, add what you want to monitor by clicking the Add button. When you're done adding all of the objects or counters, click Close to return to the Counter Log's properties window.
6. Specify the snapshot interval (the default is 15 seconds).
7. On the Log Files tab, you can specify the location of the log files or the name of the log file by clicking the Configure button. Also, you can specify how to end log filenames, the file format of the log file, and any log file size restrictions. See the following "Log Files Tab" section for more information.
8. The Schedule tab lets you specify more options for starting and stopping the log file. You can also specify the action to take when the log file closes. See the "Schedule Tab" section later in this chapter for more information.

Log Files Tab As mentioned in Step 7 in the preceding list, the Log Files tab (see Figure 25-6) offers many options. These options are important because they not only can affect your monitoring methodology, but some can also affect your system's performance.

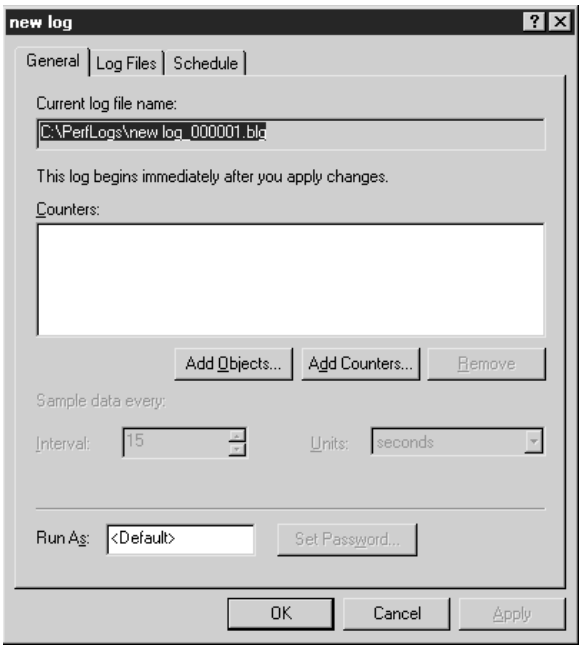


Figure 25-5. *Adding objects or counters to a log file*

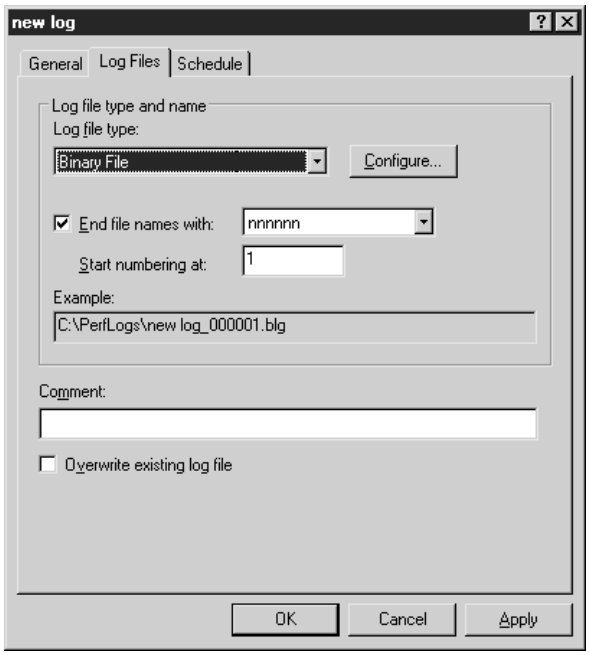


Figure 25-6. *The Log Files tab*

Log File Location By default, Performance Logs and Alerts stores all log files in the %Systemroot%\PerfLogs directory. This directory should be immediately changed to another disk drive. Both Windows Server 2003 and monitoring processes are competing against one another for resources by using the same drive. Moving the entire disk I/O associated with counter log writes to the drive frees valuable resources and enables your system to function more efficiently.

End File Names A convenient way to keep track of log files is to end the log filename with a number or a date. The default configuration enables this feature, and we highly recommend that you use this naming mechanism, especially if you're considering creating sequential log files.

File Type Log files can now be saved in two types of text format, two types of binary format, as well as in a SQL database. Table 25-2 list these formats and gives a brief description of each.

Log File Size The log file size setting enables you to control the growth of log files. Here are some of the benefits of using this feature:

- Controlling the size of the log files makes them easier to manage.
- You reduce the chances of running out of disk space.
- Limited data collection is easier to analyze.

Note *Windows Server 2003 supports log file sizes beyond 1GB. You may also append these log files to keep performance data contiguous.*

Schedule Tab As mentioned earlier, the Schedule tab (see Figure 25-7) has a variety of options that control the starting and stopping of the log file as well as options that become effective when the log file stops.

Of course, you can always manually start and stop a log file using the CD-player-like buttons located in the Counter Logs pane, but the real advantage of the Schedule tab comes when you configure automatic start and stops. In the Stop log section, you can specify when to stop the log in seconds, minutes, hours, or days. You can also opt to stop the log file at a specific time and date, or when it reaches a specified capacity if using the Binary Circular File format.

Also located in the Stop log section of the Schedule tab are options to start a new log when a log file closes and to run a command.

Log File Format	Description
Text File – CSV	Comma-delimited file. This file format can easily be read by spreadsheets such as Microsoft Excel.
Text File – TSV	Tab-delimited file. This file format is suitable for viewing with spreadsheet and database programs.
Binary File	This format refers to a sequential, binary format that uses the .blg extension. Use this file format when creating multiple sequential logs.
Binary Circular File	This format refers to a circular, binary format that uses the .blg extension. After the log file reaches its capacity, it will begin to overwrite data starting from the beginning of the file.
SQL Database	Storing performance data in a SQL database can be an extremely useful format for retrieving the specific information that you need. Also, it should be used when monitoring multiple computers and collecting large amounts of data.

Table 25-2. *Logging Formats*

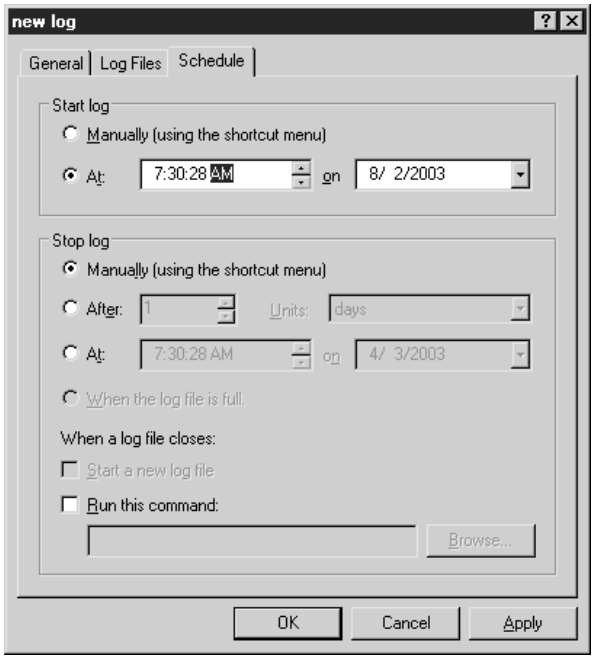


Figure 25-7. *The Schedule tab*

Creating Sequential Logs You can now more easily manage log files by creating sequential logs. Sequential logs are useful because you can keep separate log files for specified periods of time. For instance, if you want to create a log file for each day of the week, you can easily do so by creating sequential logs.

There are a few important points to remember when you configure Performance Logs and Alerts to create sequential logs:

- Use the End file names with check box on the Log Files tab and choose the Numbers format unless you're confident that the maximum capacity you specified won't be reached.
- Use the binary format.
- On the Schedule tab, at the bottom of the Stop log section, choose Start a new log file.

Working with Trace Logs

Trace logs record data when an event from the operating system or an application occurs. Events are classified as either system provider events or non-system provider events. Examples of system provider events include, but aren't limited to, the following:

- Hard disk I/O
- Process creations and deletions
- Thread creations and deletions
- TCP/IP errors
- Page faults

Trace logs differ from counter logs in the type of data they collect as well as in the frequency of the data collection. Trace logs monitor events continuously instead of at intervals.

The process of creating a trace log is very similar to that for creating a counter log, explained earlier in the "Working with Counter Logs" section. To create a trace log, click Trace Log under Performance Logs and Alerts in the left pane of the Performance window, and then right-click in the right pane and select New Log Settings. Name the log file; the trace log properties window appears.

In the trace log properties window, shown in Figure 25-8, notice the similarities with the counter log properties window. The General tab is slightly different, and there is an additional tab called Advanced. The Log Files and Schedule tabs are identical to the ones for the counter log.

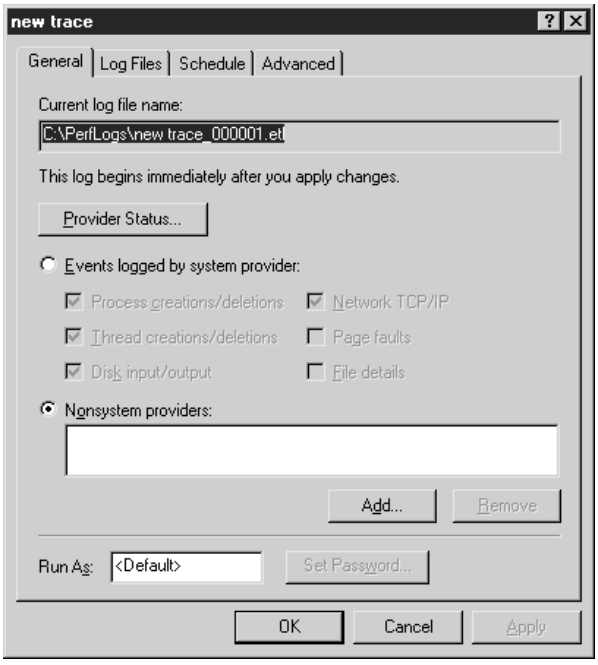
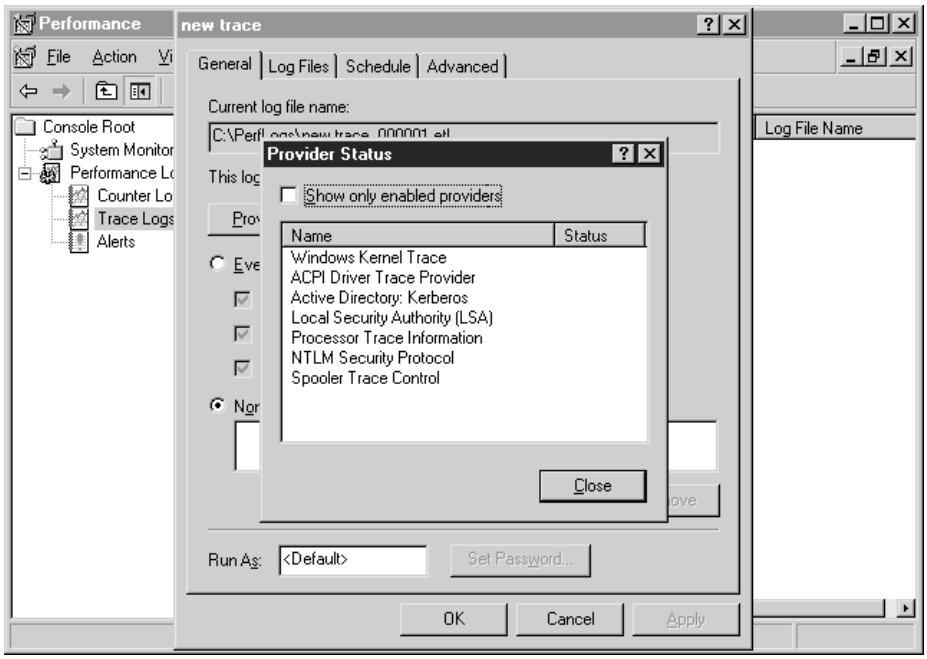


Figure 25-8. *Trace log properties window*

General Tab At the top of the General tab, you can see that the log filename ends in .etl. You then see the available system and non-system providers that you can monitor. By selecting Events logged by system provider, you can choose events by selecting the check boxes beside them. The Page faults and File details events are not checked by default because they tend to produce a tremendous amount of data. If you plan to monitor these events, Microsoft recommends monitoring them for a maximum of two hours at a time.

Click the Provider Status button to display a list of the current providers and their status (enabled and running or stopped).



Advanced Tab The Advanced tab, shown in Figure 25-9, lets you configure buffer settings. Data that is being logged is first transferred to memory buffers before the data is written to the trace log. By default, the buffers are filled to capacity before the data is written to the log file. In most scenarios, it is recommend to keep the default settings.

Viewing Log Files with System Monitor

Once you have a log file containing raw system performance data, you can retrieve and analyze the data that has been collected. To view a log file, follow these steps:

1. In the left pane of the Performance snap-in window, select System Monitor.
2. In the right pane, right-click and select Properties.

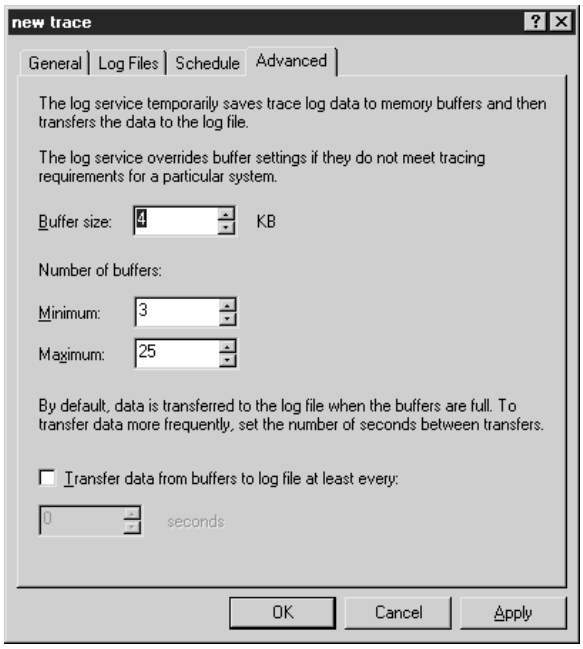


Figure 25-9. *Trace log Advanced tab*

3. On the Source tab, either click Log files and then click Add to locate the log file you want to view or click Database to specify a SQL database. Click OK when you've selected the log file.
4. You can either click OK to view the entire log file or click the Time Range button to specify the time you want to view.
5. At this point, you can also switch to the Data tab and select the counters you want to view.

Working with Alerts

Alerts can be set on any available counter to notify the administrator when a specified condition occurs, such as when processor use exceeds 90 percent. If a counter exceeds or falls below the value that you specify, the Performance Logs and Alerts service triggers an alert that logs the event and can also trigger another event, such as sending a notification message, starting a performance data log, or running a program.

Note *Make sure that the Alerter service is running before trying to configure an alert log. Also, if you plan to send notifications when an alert is triggered, make sure that the Messenger service is started.*

To create an alert, do the following:

1. In the left pane of the Performance snap-in window, click Alerts under Performance Logs and Alerts.
2. Right-click in the right pane and select New Alert Settings.
3. Name the alert and click OK.
4. On the General tab, you can optionally add a comment to identify the alert.
5. Click Add to display the Add Counters window, and add the counters that you want to monitor. Click Close when you're done.
6. For each counter that you want to monitor, specify the condition that will trigger an alert, as shown in Figure 25-10.
7. Set the snapshot interval (the default is every 5 seconds).

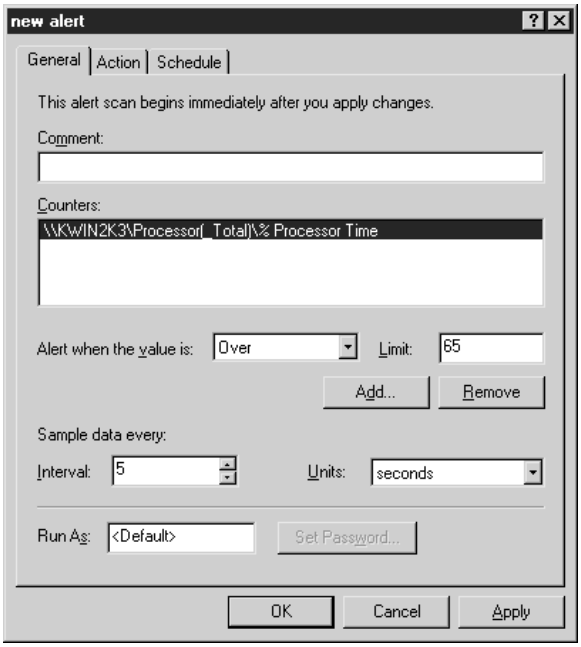


Figure 25-10. *Specifying alert conditions*

8. Select the Action tab and choose the events that will occur when an alert is triggered (see Figure 25-11). Note that any action specified here will apply to all counters being monitored in this alert log. If you want to have different actions for different counters, you'll need to create separate alert logs.

Select the Schedule tab to modify the start and stop times for alert logging.

Third-Party Utilities

In addition to the Microsoft tool set, a number of third-party capacity planning utilities are available for Windows Server 2003. Some of these tools are listed in Table 25-3.

These products commonly provide a means for collecting, analyzing, storing, and reporting statistical system information much as Windows Server 2003's Performance snap-in does. Most, if not all, of the products also incorporate enhancements such as scheduling or graphical reporting capabilities. Some even integrate innovative functionality that promises to automate many aspects of performance optimization. For example, some of the more advanced programs, such as PATROL, perform historical trend analysis and incorporate decision-support models to help you predict future system use.

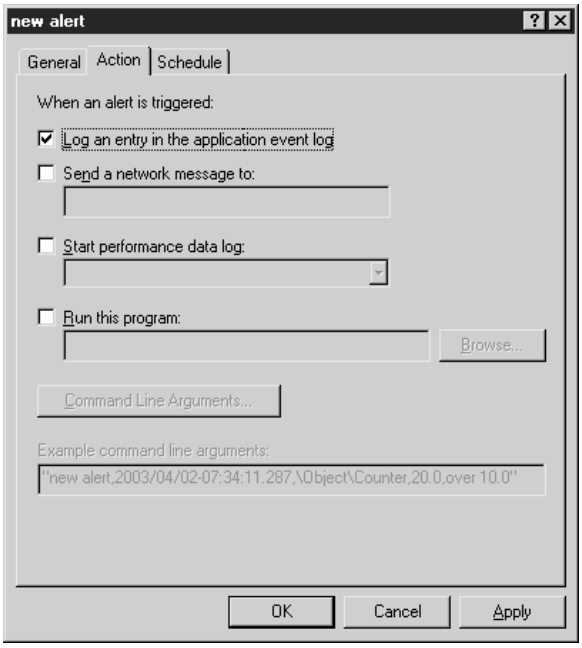


Figure 25-11. *Specifying the action that occurs when an alert is triggered*

Utility Name	Company
HP OpenView	Hewlett Packard Web site: www.openview.hp.com/
Unicenter TNG	Computer Associates Web site: www.cai.com/unicenter/
PerfMan	Information Systems Web site: www.infosysman.com/
PATROL	BMC Software Web site: www.bmc.com/products/

Table 25-3. *Third-Party Monitoring Tools*

Whether third-party products add enhanced storage features or GUI enhancements, most are superior in overall functionality to Windows Server 2003's Performance snap-in. However, there are advantages (for example, trend analysis, ease of use, and reporting) and disadvantages (such as cost and complexity) to using these utilities instead of the free, built-in utility.

Monitoring and Optimizing System Resources

You can monitor numerous system resources for the purpose of performance optimization. In fact, there are so many objects and counters that you can monitor that you can quickly become overwhelmed with the amount of data that you collect. If you do not carefully choose what to monitor, you may collect so much information that the data will be of little use. Large amounts of data can be unwieldy and can cause you to spend most of your time organizing instead of analyzing. Keep in mind that one of the key concepts behind capacity planning is *efficiency.* Tailor your monitoring to the server's configuration as accurately as possible.

There are a few important resources that you should always monitor for every server: the memory, processor, disk subsystem, and network subsystem. These resources are the four most common contributors to system bottlenecks. A *bottleneck* is the slowest component of your system and can be either hardware or software. Bottlenecks limit a system's performance because your system runs only as fast as its slowest resource. For example, a file server may be equipped with a gigabit network interface card (NIC), but if the disk subsystem is relatively antiquated, the system cannot take full advantage of the network throughput provided by the NIC. There are also residual effects of bottlenecks, such as the underconsumption of hardware resources. Resources may not be utilized because the system is trying to compensate for the bottleneck.

In addition, the way a Windows Server 2003 server is configured functionally influences the resources or services that you should consider monitoring. For example, the most common Windows Server 2003 configurations enable database, file, and print sharing, application sharing, domain controller functions, and a number of other functions. You may want to monitor the effects of replication and synchronization on domain controllers, but not for an application for file and print servers. It is important to monitor the most common contributors to system bottlenecks as well as those that pertain to the particular server configuration.

This section discusses specific counters you should monitor for each common contributor to bottlenecks. Note, however, that there are many other counters that you should consider monitoring in addition to the ones described here. This section is intended to give you a baseline or an absolute minimum number of counters to start your monitoring process.

Monitoring Memory

Of the four common contributors to bottlenecks, memory is usually the first resource to cause performance degradation. This is simply because Windows Server 2003 tends to devour memory. Fortunately, adding more memory is also the easiest and most economical way to upgrade performance. Figure 25-12 shows System Monitor's screen for monitoring memory counters in real time.

Memory has many significant counters associated with it. However, the two counters that should always be monitored are Page Faults/sec and Pages/sec. These indicate whether the system is configured with the proper amount of RAM.

A page fault occurs when a process requires code or data that is not in its *working set*. A working set is the amount of committed memory for a process or application. The Page Faults/sec counter includes both hard faults (those that require disk access) and soft faults (where the faulted page is found elsewhere in memory). Most systems can handle a large number of soft faults without sacrificing performance. However, hard faults can cause significant delays because of hard disk access times. Even the seek and transfer rates of the fastest drive available on the market are slow compared to memory speeds. The enormous latency associated with hard page faults should immediately convince you to configure the system with as much RAM as possible.

The Pages/sec counter reflects the number of pages read from or written to disk to resolve hard page faults. Hard page faults occur when a process requires code or data that is not in its working set or elsewhere in memory. The code or data must be found and retrieved from disk. This counter is the primary indicator of *thrashing* (relying too much on the hard disk drive for virtual memory) and excessive paging. Microsoft states that if the Pages/sec value is consistently above 5, you should suspect that your system may have insufficient memory. When this value is consistently above 20, you may begin to notice slower performance because of insufficient memory.

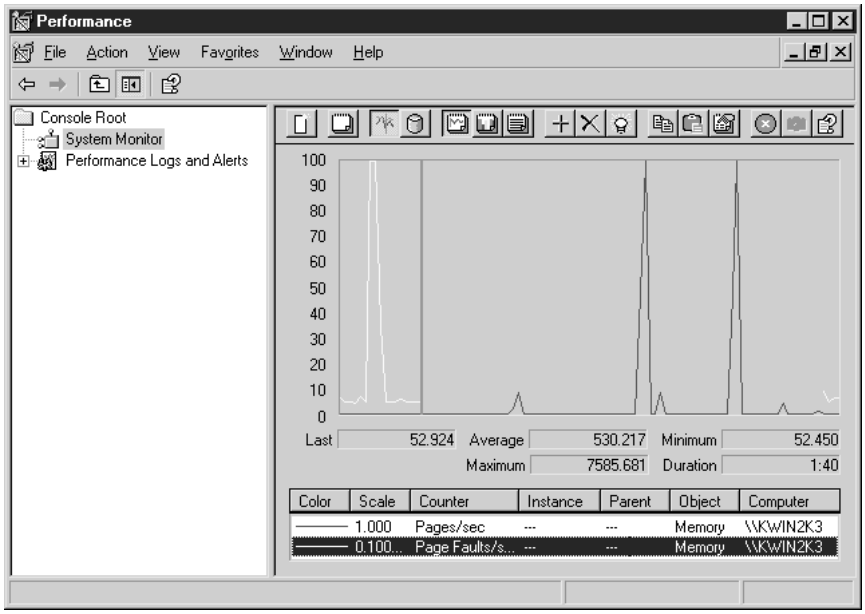


Figure 25-12. *System Monitor monitoring memory counters in real time*

Monitoring the Processor

The processor is often the first resource analyzed when there is a noticeable decrease in system performance. For performance optimization purposes, there are two significant counters to monitor in the processor object: % Processor Time and Interrupts/sec. The % Processor Time counter indicates the percentage of overall processor utilization. If more than one processor exists on the system, an instance for each one is included along with a total (combined) value counter. If the % Processor Time counter sustains a processor use rate of 50 percent or greater for long periods of time, you should consider upgrading. When the average processor time consistently exceeds 65 percent utilization, users may notice a degradation in performance that will not be tolerable.

The Interrupts/sec counter is also a good indicator of processor utilization. It indicates the number of device interrupts that the processor is handling per second. The device interrupt can be hardware or software driven and can reach high values into the thousands. Some ways to improve performance include off-loading some services to another, less-used server, adding another processor, upgrading the existing processor, clustering, and distributing the load to an entirely new machine.

Monitoring the Disk Subsystem

The disk subsystem consists of two main types of resources: hard disk drives and hard disk controllers. The Performance snap-in does not have an object directly associated with the hard disk controller because the values given in the Physical and Logical Disk objects accurately represent disk subsystem performance.

Note *Both the Physical and Logical Disk objects are enabled by default.*

Today, virtually every system component is more powerful than ever, and this is true for components within the disk subsystem as well. As a result, the effects of disk subsystem performance objects are becoming increasingly negligible and, depending on your system configuration, perhaps even unnoticeable.

Windows Server 2003 also gives you flexibility in starting and stopping disk subsystem objects. You can use **diskperf -y** to enable disk counters, **diskperf -y \\mycomputer** to enable them on remote machines, or **diskperf -n** to disable them just as you could prior to Windows Server 2003. Where the flexibility comes in is in the ability to enable the Logical Disk and Physical Disk objects separately. To specify the object that you want to activate or deactivate, include a **d** for the Physical Disk object or a **v** for the Logical Disk object. For instance, to begin viewing Logical Disk statistics, you must re-enable the Logical Disk performance object with the command **diskperf –yv**.

The best, but certainly not necessarily the only, disk performance counters to monitor for performance optimization are % Disk Time and Avg. Disk Queue Length. The % Disk Time counter monitors the amount of elapsed time that the selected physical or logical drive spends servicing read and write requests. Avg. Disk Queue Length indicates the number of outstanding requests (requests not yet serviced) on the physical or logical drive. This value is an instantaneous measurement rather than an average over a specified interval, but it still accurately represents the number of delays the drive is experiencing. The request delays experienced by the drive can be calculated by subtracting the number of spindles on the disk from the Avg. Disk Queue Length measurement. If the delay is frequently greater than 2, then the disks are degrading performance.

Monitoring Network Performance

Because of its many components, the network subsystem is one of the most complicated subsystems to monitor for bottlenecks. Protocols, NICs, network applications, and physical topologies all play important roles in your network. To further complicate matters, your environment may implement multiple protocol stacks. Therefore, the network performance counters you should monitor vary depending upon your system's configuration.

The important information to gain from monitoring network subsystem components is the amount of network activity and throughput. When monitoring network subsystem components, you should use other network monitoring tools in addition to the

Performance snap-in. For example, consider using Network Monitor (either the built-in or SMS version) or a systems management application such as MOM. Using these tools together broadens the scope of monitoring and more accurately represents what is occurring within your network infrastructure.

This discussion of performance optimization for the network subsystem focuses on TCP/IP. Saying that Windows Server 2003 relies heavily on this protocol is an understatement. The counters for TCP/IP are added to the system after the protocol is installed and include counters for Internet Protocol version 6 (IPv6).

There are many significant counters within the objects related to TCP/IP that you should consider monitoring. Two important counters to use for TCP/IP monitoring pertain to the NIC object. They are the Bytes Total/sec and the Output Queue Length counters. The Bytes Total/sec counter indicates the amount of inbound and outbound TCP/IP traffic experienced by your server. The Output Queue Length counter indicates whether there are congestion or contention problems on your NIC. If the Output Queue Length value is consistently above 2, check the Bytes Total/sec counter for abnormally high values. High values for both counters suggest that there is a bottleneck in your network subsystem, and it may be time to upgrade your server network components.

There are many other counters that need to be monitored and consulted before you can accurately pinpoint the cause of abnormal counter values or network performance degradation. For example, were the abnormal Bytes Total/sec and Output Queue Length values the result of a temporary burst in network activity or unusually high collision rates? If you know that the collision rate is greater than 10 percent, then the problem may be the performance of the overall network and not just the Windows Server 2003 server in question.

Controlling System Resources

Throughout this chapter, we've analyzed various ways to monitor and utilize system performance data. Although monitoring or analyzing performance data is necessary to more accurately tweak or otherwise tune system performance, it does not provide a direct way to control the resources that you're monitoring. As such, Microsoft has developed an MMC snap-in called Windows System Resource Monitor (WSRM) to give that level of control in today's systems.

Windows System Resource Monitor

WSRM, shown in Figure 25-13, is a utility that can be used with Windows Server 2003 Enterprise or Datacenter Editions. It gives you additional control over system resources and processes. You can use WSRM to control applications, services, and process resource utilization (e.g. processor utilization, memory usage, and processor affinity).

To control or protect system resources and utilization, ceiling values for applications, services, or processes are set using policies. These policies are customizable so that you can easily apply different standards to different systems. In addition to setting the

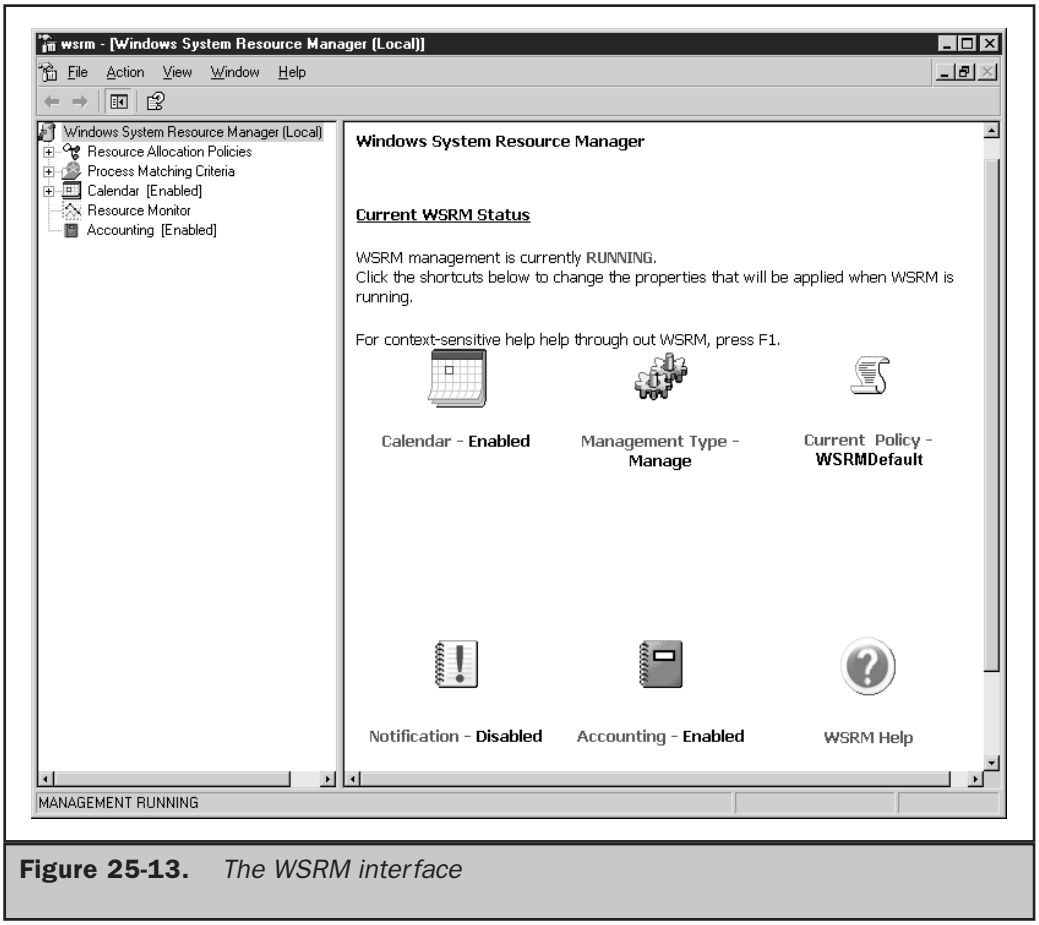


Figure 25-13. *The WSRM interface*

utilization limitations, you also can take into account scheduling considerations. So, for example, you can limit a specific application to using only 25 percent of processor utilization during peak hours of the business day. WSRM manages its own scheduling through the built-in calendar function.

WSRM is especially useful for systems serving multiple functional roles (gone are the days when you should install and configure only one type of workload or role per system). The different functionalities may compete for system resources in order to complete the tasks at hand. If an application, service, or process associated with a particular server role is susceptible to dominating system resources, WSRM can step in and ensure that the policy boundaries are not exceeded.