
Module 2: Preparing to Monitor Server Performance

Contents

Overview	1
Lesson: Introduction to Monitoring Server Performance	2
Lesson: Performing Real-Time and Logged Monitoring	7
Lesson: Configuring and Managing Counter Logs	20
Lesson: Configuring Alerts	34
Lab A: Preparing to Monitor Server Performance	42



Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2003 Microsoft Corporation. All rights reserved.

Microsoft, MS-DOS, Windows, Windows NT, Active Directory, ActiveX, JScript, MSDN, PowerPoint, Visual Basic, Visual C++, Visual InterDev, and Windows Media are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Overview

- 
- Introduction to Monitoring Server Performance
 - Performing Real-Time and Logged Monitoring
 - Configuring and Managing Counter Logs
 - Configuring Alerts

Introduction

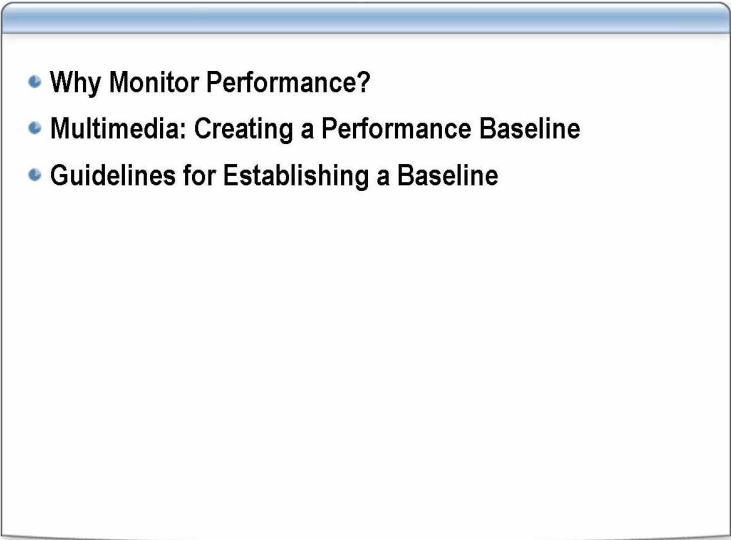
Monitoring server performance is an important part of maintaining and administering your operating system. Routine performance monitoring ensures that you have up-to-date information about how your computer is operating. Performance monitoring also provides you with data that you can use to predict future growth and to plan for how changes to your system configurations may affect future operation.

Objectives

After completing this module, you will be able to:

- Establish a performance baseline.
- Perform real-time and logged monitoring.
- Configure and manage counter logs.
- Configure alerts.

Lesson: Introduction to Monitoring Server Performance

- 
- Why Monitor Performance?
 - Multimedia: Creating a Performance Baseline
 - Guidelines for Establishing a Baseline

Introduction

This lesson explains the concept of performance monitoring, a baseline, performance objects, and counters. It also describes how to establish a performance baseline.

Lesson objectives

After completing this lesson, you will be able to:

- Explain the reason for monitoring performance.
- Explain what a baseline is and when to create one.
- Describe a performance object.
- Explain a counter.
- Explain the guidelines for establishing a baseline.

Why Monitor Performance?

- **By monitoring performance, you obtain data that you can use to:**
 - Understand your workload and the corresponding effect on your system's resources
 - Observe changes and trends in workloads and resource usage so you can plan for future upgrades
 - Test configuration changes or other tuning efforts by monitoring the results
 - Diagnose system problems and identify components or processes for optimization
- **Analyze performance data to uncover bottlenecks**

Introduction

Monitoring performance is a necessary part of preventive maintenance for your server. By routinely monitoring the performance of your server over periods ranging from days to weeks to months, you can establish a baseline for server performance. Through monitoring, you obtain performance data that is useful in diagnosing server problems.

Why monitor performance?

You use performance data to:

- Understand your workload characteristics and the corresponding effect on your system's resources.
- Observe changes and trends in workload characteristics and resource usage so you can plan for future upgrades.
- Test configuration changes or other performance tuning efforts by monitoring the results.
- Diagnose problems and identify components or processes for optimization.

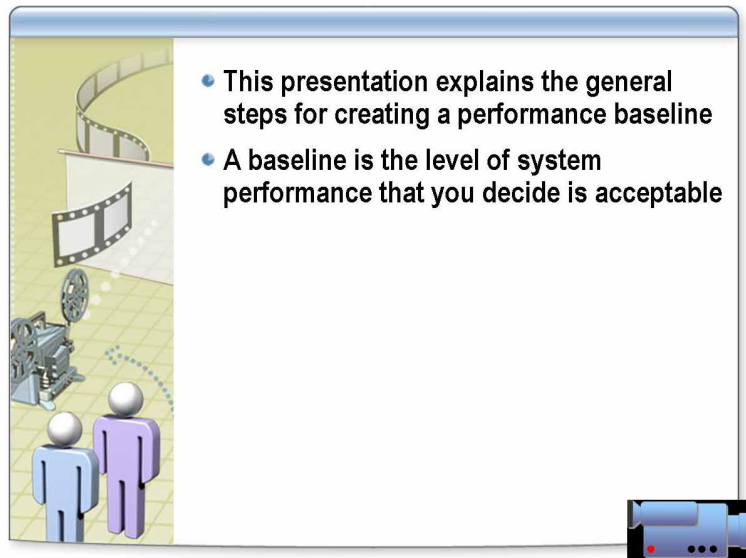
Analysis of performance data and bottlenecks

Analysis of performance data can reveal problems, such as excessive demand on certain resources that results in bottlenecks. A bottleneck exists when a single resource adversely affects the performance of the whole system. Demand on the single resource may become excessive enough to cause a bottleneck of the four subsystems: memory, processor, disk, and network.

Some of the reasons that bottlenecks occur are:

- Subsystems are insufficient, so additional or upgraded components are required. For example, lack of memory is a major cause of bottlenecks.
- Subsystems are not sharing workloads evenly and need to be balanced. For example, an older network card that is installed on a new server may cause a bottleneck.
- A subsystem is malfunctioning and needs to be replaced. For example, a hard disk often has minor problems before it fails.
- A program is monopolizing a particular resource. For example, a custom program that was written by a consultant may not be sharing memory correctly. Solutions to this problem include substituting another program, asking a developer to rewrite the program, adding or upgrading resources, or running the program during periods of low demand.
- A subsystem is incorrectly configured, so configuration settings must be changed. For example, an older multispeed network card may be configured for 10 megabits per second (Mbps) when it should be set to 100 Mbps.

Multimedia: Creating a Performance Baseline



File location

To view the *Creating a Performance Baseline* presentation, open the Web page on the Student Materials compact disc, click **Multimedia**, and then click the title of the presentation.

Objectives

After completing this presentation you will be able to:

- Explain the purpose of a baseline.
- Describe how to use the Performance console.

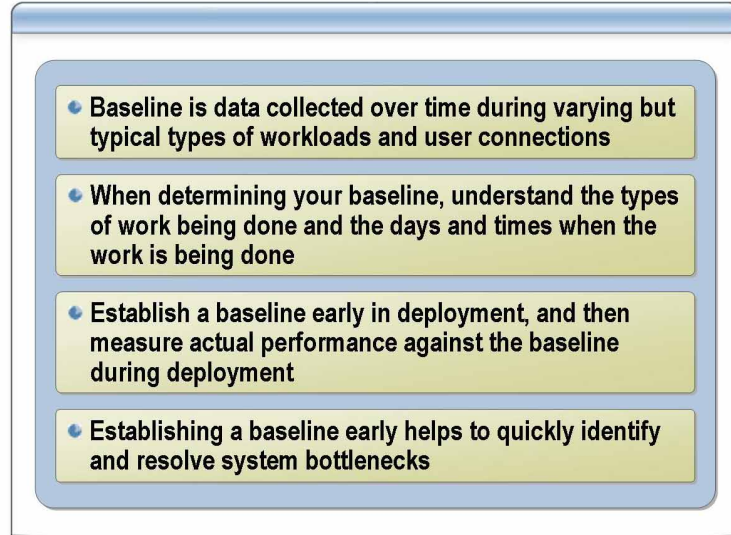
Key points

Key points from the presentation are summarized in the following list:

- Baseline
 - Take samples of counter values every 30 to 45 minutes for a week, during peak, low, and normal operations.
- General steps for creating a baseline
 - a. Identify resources
 - b. Capture data
 - c. Store data
- Four major system resources for performance baselines
 - Memory
 - Processor
 - Physical disk
 - Network
- Performance object

A performance object is the data generated by a system component or resource. Each performance object provides counters, which represent data about specific aspects of system performance. Performance objects can have multiple instances.

Guidelines for Establishing a Baseline



Introduction

You derive a baseline measurement from a collection of data over an extended period, during varying but typical types of workloads and user connections. The baseline is an indicator of how individual system resources or a group of resources are used during periods of normal activity.

Factors to consider when determining a baseline

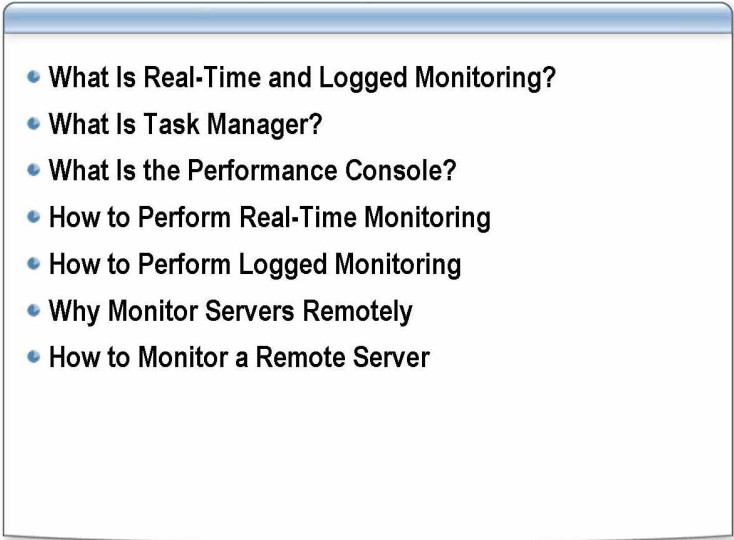
You should consider the following factors when determining a baseline:

- When you determine your baseline, it is important to know what type of work is being performed and when it is being performed. That information helps you to associate specific work with specific resource usage and to determine whether the level of performance during those intervals is reasonable. After you gather performance data over an extended period of low, average, and peak usage, you can determine what constitutes acceptable performance for your system. That determination is your baseline.

For example, if performance diminishes briefly at a certain time of day, and you find that many users log on or off at that time, the slowdown may be acceptable. Similarly, if performance is poor every evening at a certain time when no users are logged on to the system but nightly backups are being performed, the diminished performance may be acceptable. You can determine what performance level is acceptable only when you know the degree of performance loss and its cause.

- Establish a baseline early in the deployment phase. Then, during deployment, you can measure the baseline against actual performance.
- Establishing a baseline early helps you to quickly identify and resolve system bottlenecks.
- Use your baseline to watch for long-term changes in usage patterns that require increased capacity.

Lesson: Performing Real-Time and Logged Monitoring

- 
- What Is Real-Time and Logged Monitoring?
 - What Is Task Manager?
 - What Is the Performance Console?
 - How to Perform Real-Time Monitoring
 - How to Perform Logged Monitoring
 - Why Monitor Servers Remotely
 - How to Monitor a Remote Server

Introduction

The primary monitoring tools in Microsoft® Windows® Server 2003 are the Performance console and Task Manager.

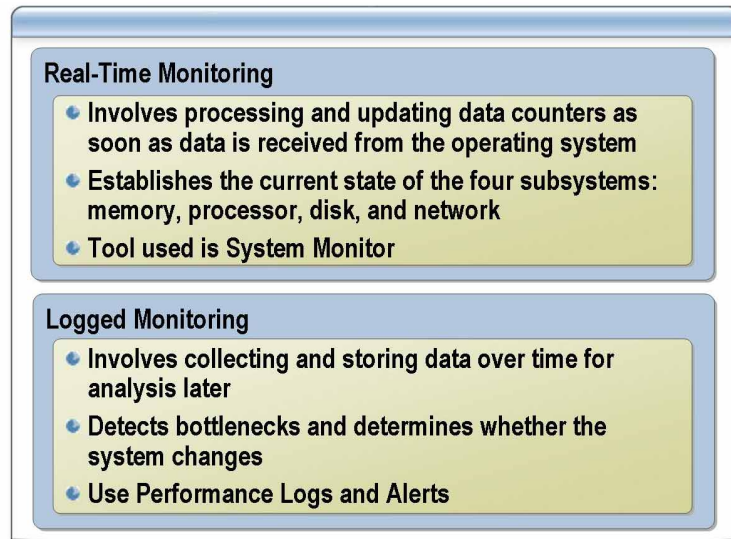
This lesson describes how to perform monitoring by using Performance and Task Manager.

Lesson objectives

After completing this lesson, you will be able to:

- Explain real-time monitoring and logged monitoring.
- Describe the Task Manager tool.
- Describe the Performance console.
- Perform real-time monitoring by using Task Manager and Performance.
- Perform logged monitoring by using Performance.
- Explain the reasons for monitoring remote servers from a workstation.
- Use Performance to monitor a remote computer.

What Is Real-Time and Logged Monitoring?



Introduction

Administrators can use logged monitoring to monitor servers on a continuous basis. By configuring logged monitoring, you can establish a performance baseline and use trend analysis to identify server problems. For example, if users complain that an application server is gradually slowing down, you can check the log files for that server to investigate the cause of the problem.

Administrators also must investigate problems caused by specific events. For this type of problem, you must enable real-time monitoring. For example, if a help desk technician tells you that the printers attached to the print server are printing intermittently, you use a real-time monitor, such as Task Manager or System Monitor, to investigate the cause of the problem.

Real-time monitoring

In real-time monitoring, System Monitor processes and updates data counters as soon as the data is received from the operating system. You use real-time monitoring to establish the current state of the four subsystems: memory, processor, disk, and network. For example, if users complain about the slow response time of a client/server application in a situation that caused no previous problems, you can use System Monitor to diagnose and troubleshoot the problem.

Logged monitoring

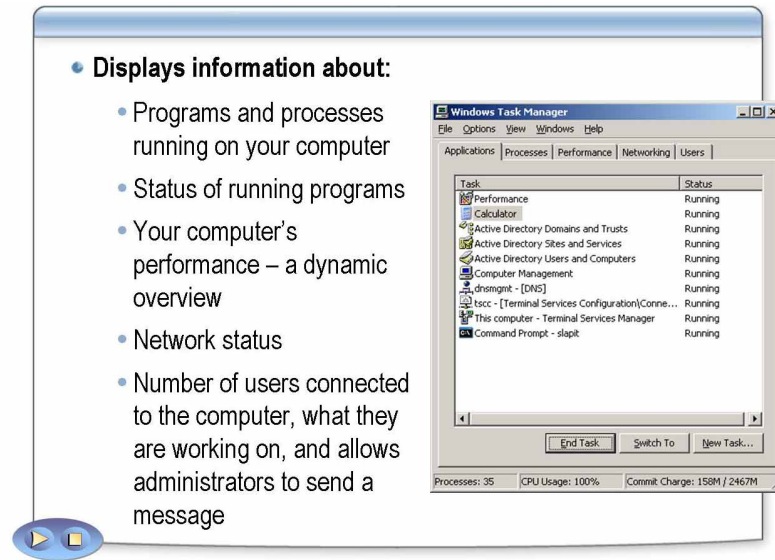
Logged monitoring involves collecting and storing data over time for analysis later. Use logged monitoring to establish a baseline, detect bottlenecks and determine whether the system changed over time. Use the Performance Logs and Alerts tool for logged monitoring.

Example of logged monitoring

For example, when your organization acquires a new server, to understand its capabilities, you can configure several counters to determine memory usage, CPU usage, disk usage, and network usage. You can use the data that you collect to determine the range of counter values that are normal for your environment.

You can also set up multiple logs to monitor several events at various times. This way, you can determine whether events such as backup, domain replication, or users who connect remotely on evenings and weekends cause bottlenecks on the server.

What Is Task Manager?



Introduction

Task Manager provides an overview of system activity and performance. It provides information about programs and processes that are running on your computer. It also displays the most commonly used performance measures for processes. You can use Task Manager to perform real-time monitoring.

Task Manager functions

You can use Task Manager to monitor key indicators of your computer's performance:

- You can see the status of the programs that are running and end programs that are not responding.
- You can also assess the activity of running processes by using up to fifteen parameters, and view graphs and data about CPU and memory usage.
- If you are connected to a network, you can view network status.
- If more than one user is connected to your computer, you can see who is connected, see what files they are working on, and send them a message.

Task Manager has five tabs that allow you to perform all these functions.

Applications tab

The **Applications** tab displays the status of the programs that are running on the computer. On this tab, you can end, switch to, or start a program.

Processes tab

The **Processes** tab displays information about the processes that are running on the computer. For example, you can display information about CPU and memory usage, page faults, handle count, and other parameters.

Performance tab

The **Performance** tab displays a dynamic overview of your computer's performance, including:

- Graphs of CPU and memory usage.
- The number of handles, threads, and processes that are running on your computer.
- The amount, in kilobytes, of physical, kernel, and commit memory. Physical memory is total memory, kernel memory is the memory that the system kernel and device drivers use, and commit memory is the amount of memory that is allocated to programs and the operating system.

Networking tab

The **Networking** tab displays a graphical representation of network performance. It provides a simple, qualitative indicator that shows the status of the network or networks that are running on your computer. The **Networking** tab is displayed only if a network card is present.

On this tab, you can view the quality and availability of your network connection, whether you are connected to one or more networks.

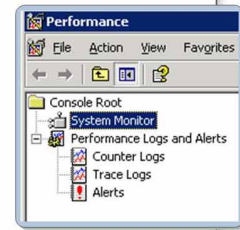
Users tab

The **Users** tab displays the names of users who can access the computer, along with session status and names. **Client Name** specifies the name of the client computer that is using the session, if applicable. **Session** provides a name for you to use when you perform such tasks as sending another user a message or connecting to another user's session.

The **Users** tab is displayed only if Fast User Switching is enabled on the computer you are working on. The computer must also be either a member of a workgroup or a standalone computer. The **Users** tab is unavailable on computers that are members of a network domain.

What Is the Performance Console?

- The Performance console contains **System Monitor and Performance Logs and Alerts**
- With **System Monitor**:
 - You can collect and view real-time data of a local computer or several remote computers
 - You can create graphs, histograms, and reports of the performance counter data
- **Performance Logs and Alerts**:
 - Provides logging and alert capabilities
 - Defines settings for counter logs, trace logs, and alerts



Introduction

Windows Server 2003 provides the following tools as part of the Performance console for monitoring resource usage on your computer:

- System Monitor
- Performance Logs and Alerts

System Monitor capabilities

By using System Monitor, you can collect and view extensive data about the use of hardware resources and the activity of system services on computers that you administer.

With System Monitor you can collect and view the real-time performance data of a local computer or several remote computers.

To select the data to be collected, specify performance objects, performance counters, and performance object instances.

- A *performance object* is a logical collection of counters that is associated with a resource or service that can be monitored.
- A *performance counter* is a data item that is associated with a performance object. For each counter that you select, System Monitor displays a value that corresponds to a specific aspect of the performance that is defined for the performance object.
- *Performance object instances* are multiples of the same object type. For example, if a system has multiple processors, the Processor object type has multiple instances.

View logged counter data

You can view logged counter data by using System Monitor, or you can export the data to spreadsheet programs or databases for analysis and report generation.

By using System Monitor, you can create graphs, histograms, and reports of the performance counter data. The graph view, the default view, offers the widest variety of optional settings.

View	Description
Graph	Useful for real-time analysis of all the processes in a system Displays counter data over a given time in line graph format
Histogram	Useful for detecting processor bottlenecks Displays counter data in a bar chart, showing only one value per counter instance
Report	Useful for monitoring numerical values from each counter Displays counter data in a table, showing only one value per counter instance

Performance Logs and Alerts capabilities

Performance Logs and Alerts provide logging and alert capabilities for both local and remote computers. You use logging for detailed analysis and record-keeping. Retaining and analyzing log data that is collected over time can be helpful for capacity and upgrade planning.

Collect performance data

With Performance Logs and Alerts you can collect performance data by using two types of logs—counter logs and trace logs. You can also set an alert on a counter that sends a message, runs a program, or starts a log when the counter's value exceeds or falls below a specified setting.

Define settings

In Performance Logs and Alerts, you define settings for counter logs, trace logs, and alerts. The details pane of the console window shows counter logs and alerts that you have created. You can define multiple counter logs or alerts to run simultaneously. Each counter log or alert is a saved configuration that you define.

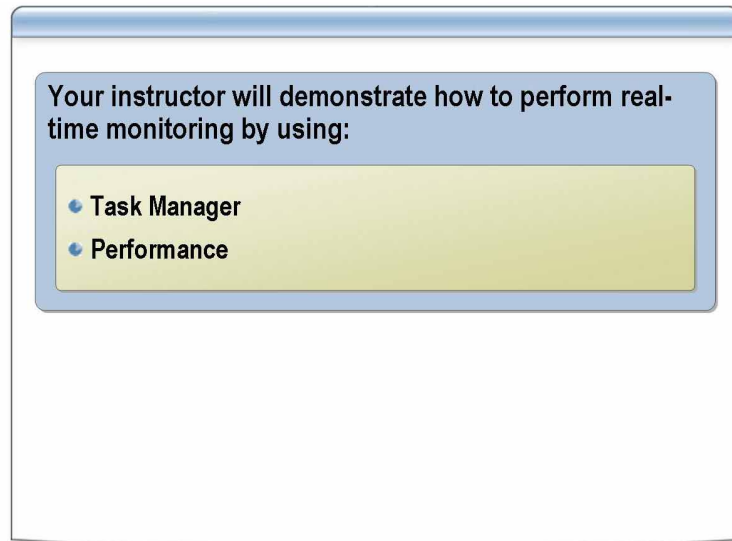
If you configure the log for automatic starting and stopping, a single log can generate many individual log data files. For example, if you generate a log file for each day's activity, one file closes at 11:59 P.M. one day, and a new file opens at midnight the next day.

Other functions

Data that Performance Logs and Alerts collects can be viewed during collection as well as after collection has stopped. Data collection occurs regardless of whether any user is logged on to the computer that is being monitored.

You can define start and stop times, file names, file sizes, and other parameters for automatic log generation, and you can manage multiple logging sessions from a single console window.

How to Perform Real-Time Monitoring



Introduction

Administrators often must perform situational real-time monitoring to answer questions about server performance from users, management, other systems administrators, and systems engineers. Task Manager is valuable when you must quickly evaluate processor usage, page file usage, and network usage. Performance monitor provides you with additional counters that can you can use to analyze problems as you view interrupts per second, queue lengths, pages per second, and so on.

Procedure for performing real-time monitoring by using Task Manager

To perform real-time monitoring by using Task Manager, press CTRL+ALT+DEL, and then click **Task Manager**.

To be monitored	Action
Applications	Click the Applications tab to monitor running applications
Processes	Click the Processes tab to monitor the running processes. On the Processes tab, click a column name to sort by that column. Click the column name a second time to reverse sort by that column. On the View menu, click Select Columns to add counters to the Processes tab.
Performance	Click the Performance tab to monitor CPU and memory usage.
Networking	Click the Networking tab to monitor network traffic to this computer.
Users	Click the Users tab to monitor the names of users who are connected to the computer.

Procedure for performing real-time monitoring by using Performance

To perform real-time monitoring by using Performance:

1. To start Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.

Note You can also start Performance by opening a command prompt window and typing **perfmon.msc**

2. Click **System Monitor**.
3. Right-click in the details pane, and then click **Add Counters**.
4. For each counter or group of counters that you want to add to the log, perform the following steps:
 - a. Under **Performance object**, select the type of performance object to monitor.
 - b. Select one of the following options to add counters:
 - **All counters**. Specifies that you want to include all counters for the selected performance object.
 - **Select Counters from list**. Specifies that you want to select individual counters for the selected performance object.

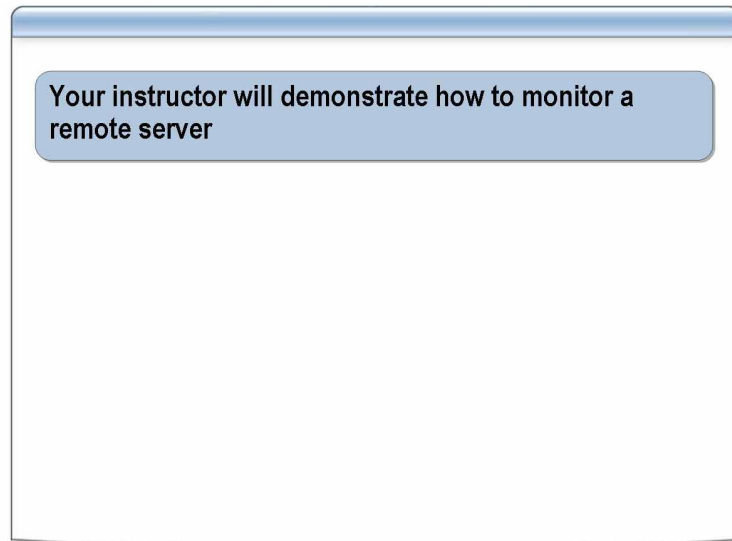
Note For a description of a counter, select the counter, and then click **Explain**.

- c. Select one of the following options to monitor the instances of the selected counters:
 - **All Instances**. Specifies that you want to monitor all instances of the selected counters.
 - **Select Instances From List**. Specifies that you want to monitor particular instances selected from the list of the selected counters.

Note Some object types have several instances. For example, if a system has multiple processors, the **Processor** object type has multiple instances. If a system has two disks, the **PhysicalDisk** object type has two instances. Some object types, such as **Memory** and **Server**, have only one instance. If an object type has multiple instances, you can add counters to track statistics for each instance, or in many cases, for all instances at once.

5. Click **Add**.
6. Click **Close**.

How to Perform Logged Monitoring



Introduction

Administrators use logged monitoring to:

- Establish a performance baseline.
- Automate monitoring.
- Capture data from multiple servers simultaneously.

Procedure

To perform logged monitoring by using Performance:

1. To start Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.
2. Double-click **Performance Logs and Alerts**.
3. Right-click **Counter Logs**, and then click **New Log Settings**.
4. In the **New Log Settings** dialog box, specify an appropriate name for the log, and then click **OK**.

5. On the **General** tab, click **Add Counters**. For each counter or group of counters that you want to add to the log, perform the following steps:
 - a. Under **Performance object**, select the type of performance object to monitor.
 - b. Select one of the following options to add counters:
 - **All counters**. Specifies that you want to include all counters for the selected performance object.
 - **Select Counters from list**. Specifies that you want to select individual counters for the selected performance object.
-

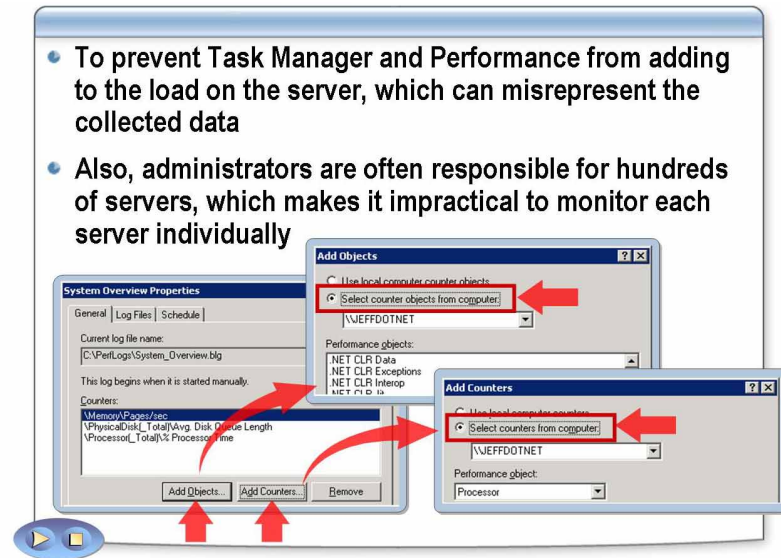
Note For a description of a counter, click the counter, and then click **Explain**.

- c. Select one of the following options to monitor the instances of the selected counters:
 - **All Instances**. Specifies that you want to monitor all instances of the selected counters.
 - **Select Instances From List**. Specifies that you want to monitor particular instances selected from the list of the selected counters.
-

Note Some object types have several instances. For example, if a system has multiple processors, the **Processor** object type has multiple instances. The **PhysicalDisk** object type has two instances if a system has two disks. Some object types, such as **Memory** and **Server**, have only one instance. If an object type has multiple instances, you can add counters to track statistics for each instance or for all instances at once.

6. Click **Add**, and then click **Close**.
7. On the **General** tab, change the **Interval** to an appropriate time.
8. On the **Schedule** tab, change the **Start log** to begin at a specific time and day, change the **Stop log** to a specific time and day, and then click **OK**.
9. If prompted to create a log folder, click **Yes**.

Why Monitor Servers Remotely?



Introduction

You are monitoring a network server that is running Microsoft SQL Server™ 2000. When you monitor the server at the console, you notice that many counters are available that do not appear when you monitor the server remotely from your workstation. This problem occurs because SQL Server is not installed on the workstation. To solve this problem, you may need to install the Management and Client tools that are available on the SQL Server compact disc on the workstation and then monitor the server remotely.

Why monitor a remote server from a workstation?

The additional load that Task Manager and Performance put on the server can cause misrepresentation of the data that you are collecting. By monitoring the server from a remote location, you reduce the likelihood of this occurring.

Also, administrators are often responsible for hundreds of servers, which makes it impractical to monitor each server individually.

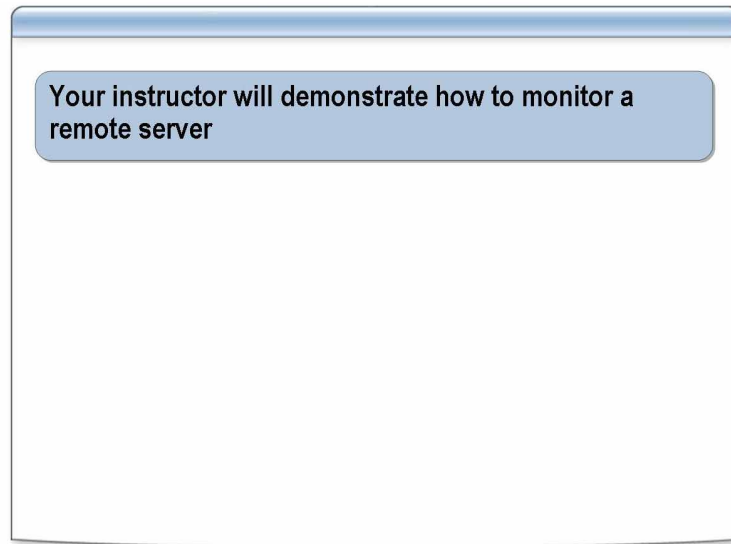
Options in the Performance console for monitoring a remote server

The Performance console provides the following options that you can use to monitor a remote server:

- You can add objects or counters to a remote server. On the **General** tab, click **Add Objects** or **Add Counters**, respectively.
- You can log objects or counters from a specific computer regardless of where the service is run. Click **Select counter objects from computer** or **Select counters from computer** respectively. Specify the name of the computer that you want to monitor remotely, such as `\\MyLogServer`.

Important If you plan to monitor remote computers, you must have been delegated the appropriate authority to gain access to them.

How to Monitor a Remote Server



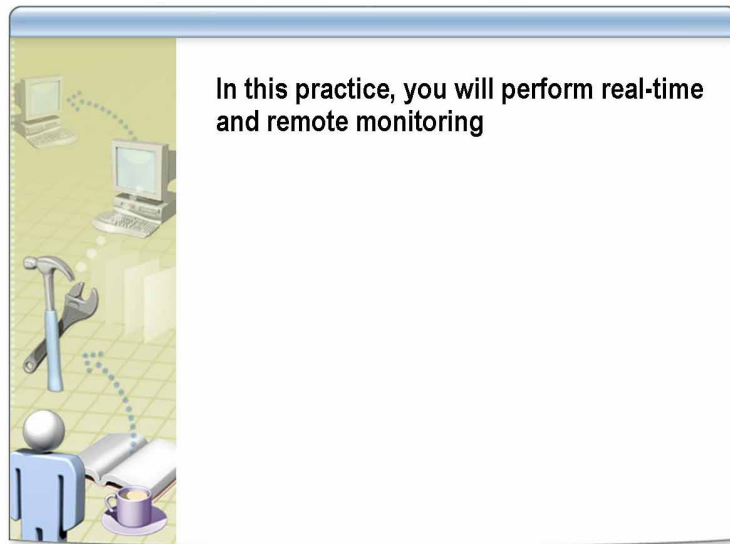
Introduction

Use the following procedure to monitor a remote server.

Procedure

1. To start Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.
2. Right-click in the right pane of System Monitor, and then click **Add counters**.
3. Click **Select counters from computer**, and then type the name of the remote computer.
4. Under **Performance Object**, in the list, select the objects that you want to monitor. For each performance object, select the appropriate counters in the list. Click **Add** each time you select a counter, and then click **Close**.

Practice: Performing Real-Time and Logged Monitoring



Objective

In this practice, you will perform real-time and remote monitoring of a server.

Scenario

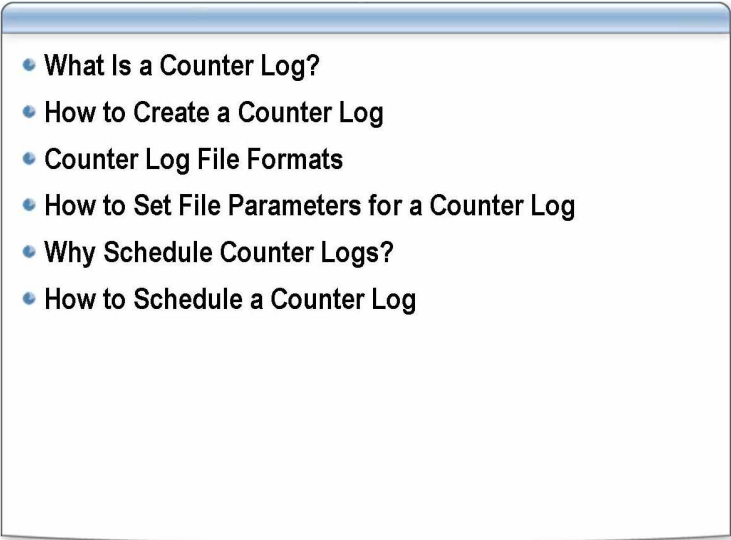
Your organization recently acquired two new servers. One of the servers is local and the second server is remote. You want to monitor both servers simultaneously from your local server by using a real-time monitoring tool.

Practice

► **Perform real-time and remote monitoring on a local and a remote computer**

1. Log on to the domain as *ComputerUser* (where *Computer* is the name of your computer) with a password of **P@ssw0rd**.
2. Open the **Run** dialog box, type **runas /env /user:nwtraders\administrator cmd** and then click **OK**.
3. When prompted for a password, type **P@ssw0rd** and press ENTER.
4. In the Command Prompt window, start **perfmon.msc**.
5. Open the **Add Counters** dialog box by clicking + and then in the **Add Counters** dialog box, select the counters from **\\GLASGOW**.
6. Add the following counters:
 - Processor\% Processor Time
 - Memory\Pages/sec
 - PhysicalDisk\Avg. Disk Queue Length
7. Close the **Add Counters** dialog box.
8. Verify that you are monitoring three counters from your server and three counters from the Glasgow server.
9. Close all windows and log off.

Lesson: Configuring and Managing Counter Logs

- 
- What Is a Counter Log?
 - How to Create a Counter Log
 - Counter Log File Formats
 - How to Set File Parameters for a Counter Log
 - Why Schedule Counter Logs?
 - How to Schedule a Counter Log

Introduction

You use counter logs to gather data about various aspects of performance objects. For example, for the **Memory** object, counter logs gather data about available memory, cache memory, and virtual memory. Counter logs are built into the operating system and continually capture data.

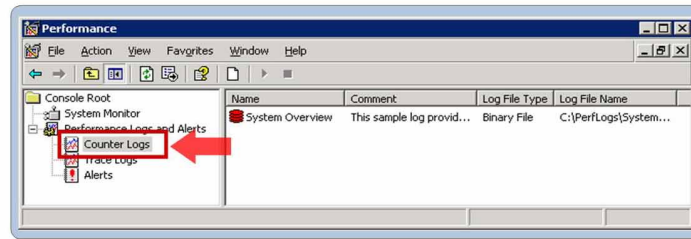
Lesson objectives

After completing this lesson, you will be able to:

- Explain a counter, counter log, and counter log data.
- Create a counter log.
- Explain the counter log file formats.
- Set file parameters for a counter log.
- Explain the reasons for scheduling a counter log.
- Schedule a counter log.

What Is a Counter Log?

- Each performance object provides performance counters that represent data about specific aspects of a system or server
- Counter logs define what data is stored in the log file



Introduction

Windows Server 2003 collects data about system resources, such as disks, memory, processors, and network components. Also, applications and services that you run on your system, such as Microsoft Exchange Server, can also collect data. This data is described as a performance object and is typically named for the component that generates the data. For example, the Processor object is a collection of performance data about the processors on your system.

Performance counter

A variety of performance objects are built into the operating system. Each performance object provides performance counters that represent data about specific aspects of a system or service. Counters are used to measure various aspects of performance. For example, the Pages/sec counter provided by the **Memory** object tracks the rate of memory paging.

Counter logs

Counter logs are counters that specify what data is stored in the log file. You use counter logs to select counters to collect performance data. You can use the Performance Logs and Alerts to create counter logs. In the interface, you select counter logs by using the **Counter Logs** option. The right pane of the Performance console window shows counter logs that you have created. You can define multiple counter logs to run simultaneously. Each counter log is a saved configuration that you define.

Counter log information in the Performance console

The following table describes the information about the counter logs that is provided by the columns in the right pane of the Performance console.

Column	Description
Name	The name of the counter log. It describes the type of data you are collecting or the condition you are monitoring.
Comment	Any descriptive information about the counter log.
Log File Type	The log file format that you define. For counter logs, this format can be binary, binary circular, text file (comma delimited), text file (tab delimited), or SQL.
Log File Name	The path and base file name that you defined for the files that are generated by this counter log. The base file name is used for automatically naming new files.

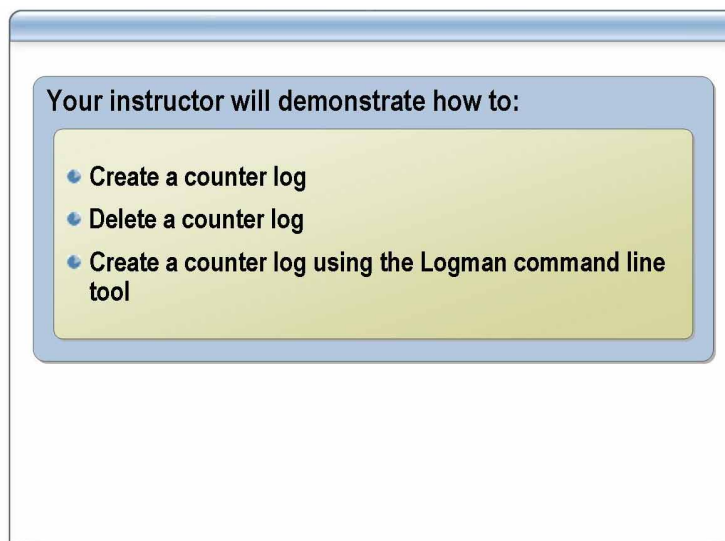
Counter log data

Counter log data is the information that you can collect automatically from local or remote computers by configuring Performance Logs and Alerts.

Counter log data:

- Can be viewed by using System Monitor.
- Can be exported to spreadsheet programs or databases for analysis and generating reports.
- Is used to compare the values against the counter thresholds to verify that resource usage or other system activity is within acceptable limits.

How to Create a Counter Log



Introduction

Counter logs record samples of data about hardware resources and system services based on performance objects, counters, and scheduled interval time.

When you create a counter log, Performance Logs and Alerts obtains data from the system when the update interval elapses. For example, when you set the counter data interval to 15 minutes, the data is collected every 15 minutes.

Procedure for creating a counter log

To create a counter log:

1. To start Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.
2. Double-click **Performance Logs and Alerts**, and then click **Counter Logs**.
Any existing counter logs are listed in the details pane. A green icon indicates that a log is running; a red icon indicates that a log is stopped.
3. Right-click a blank area of the details pane, and then click **New Log Settings**.
4. In the **Name** box, type the name of the log, and then click **OK**.
5. On the **General** tab, click **Add Counters** to select the counters that you want to log.
6. If you want to change the default file and schedule information, make the changes on the **Log Files** tab and the **Schedule** tab.

Note To save the settings for a counter log, right-click the counter log in the right pane of the Performance console, and then click **Save Settings As**. You can then specify an .htm file in which to save the settings. To reuse the saved settings for a new counter log, right-click the right pane, and then click **New Log Settings From**. This is an easy way to generate new settings from a counter log configuration. You can also open the HTML file in Microsoft Internet Explorer to display a System Monitor graph.

Procedure for deleting counter logs

Because counter logs can quickly consume a lot of storage space, delete logs when you no longer need them, usually after a baseline is established and the baseline information is recorded. A general guideline is to establish a baseline once a week and delete logs older than 30 days.

To delete a counter log:

1. To start Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.
2. Double-click **Performance Logs and Alerts**, and then click **Counter Logs**.
3. In the details pane, right-click the counter log that you want to delete.
4. Click **Delete**.

Important To perform the preceding two procedures, you must be a member of the Administrators group on the local computer, or you must have been delegated the appropriate authority. If the computer is connected to a domain, members of the Domain Admins group can perform this procedure.

As a security best practice, consider using **Run as** to perform this procedure.

Procedure for creating counter logs using the logman command line tool

You can also create counter logs using the **logman** command line tool. Logman allows you to manage and schedule performance counter log collections on local and remote servers.

To create daily performance counter queries with begin and end times, repeat collections, version control numbers, counter paths and sample intervals using logman:

1. On the **Start** menu, click **Run**, type **cmd** and then click **OK**.
2. At the command prompt, type:

```
Logman create counter daily_perf_log -b 7/27/2003 13:00:00 -e 7/27/2003 15:00:00 -r -v mmddhhmm -c "\Processor(_Total)\% Processor Time" "\Memory\Available Bytes" -si 00:15 -o "C:\perflogs\daily_log"
```

where:

-b Specifies begin-time for collections in a 24-hour format.

-e Specifies end-time for collections in a 24-hour format.

-r Repeats the collection every day at the time periods specified by the **-b** and **-e** options. This command is valid only for begin- and end-times specified on the same day, month, and year.

-v Attaches the version control information to the end of the output file and path name. Use date format *mmddhhmm* (month, day, 24-hour, minute) for version control.

-c Specifies the name of the counter

-si Specifies sample intervals for performance counter collection in hours, minutes, and seconds. Default is 15 seconds.

-o Specifies the pathname of the output file.

Additional information can be found on the **Start** menu by clicking **Help and Support** and then searching for **logman**.

Counter Log File Formats

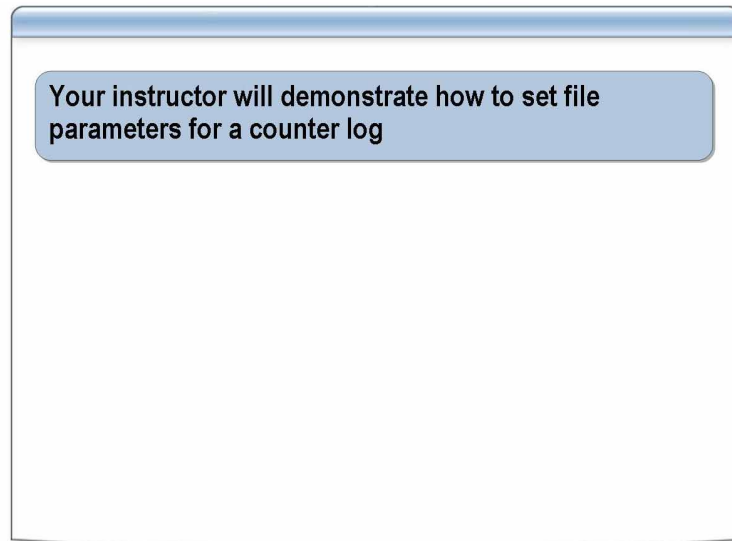
Log File Format	Description	When to use
Text File (Comma delimited)	Comma-delimited log file (with a .csv extension)	To export log data into a spreadsheet program
Text File (Tab delimited)	Tab-delimited log file (with a .tsv extension)	To export log data into a spreadsheet program
Binary File	Sequential, binary-format log file (with a .blg extension)	To record data instances that are intermittent
Binary Circular File	Circular, binary-format log file (with a .blg extension)	To record data continuously to same log file
SQL Database	Name of an existing SQL database and log set within the database where performance data will be read or written	To collect performance data at an enterprise level rather than a per-computer basis

The following table describes the log file formats that you can use to set file parameters for a counter log.

Log file format	Description	When to use
Text File (Comma delimited)	Defines a comma-delimited log file, with a .csv extension.	Use this format, for example, to export the log data into a spreadsheet program.
Text File (Tab delimited)	Defines a tab-delimited log file, with a .tsv extension.	Use this format, for example, to export the log data into a spreadsheet program.
Binary File	Defines a sequential, binary-format log file, with a .blg extension. Only binary file formats can accommodate instances that are not persistent throughout the duration of the log.	Use this file format to record data instances that are intermittent—that is, stopping and resuming after the log begins to run. Use the tracertpt command line tool to convert binary files into a comma-delimited log file.
Binary Circular File	Defines a circular, binary-format log file, with a .blg extension.	Use this file format to record data continuously to the same log file, overwriting previous records with new data when the file reaches its maximum size. Use the tracertpt command line tool to convert binary files into a comma-delimited log file.
SQL Database	Defines the name of an existing SQL database and log set within the database where the performance data will be read or written.	Use this file format to collect performance data at an enterprise level rather than on a per-computer basis.

Use the text file format or the binary file format if you must export the data to a spreadsheet program later. The binary file format is more compact than the text file format, but you must convert it to the text file format before you export it to a spreadsheet. Use the **tracert** command line tool to convert binary files into a comma-delimited log file. For example, type **tracert logfile.blg -o logfile.csv**

How to Set File Parameters for a Counter Log



Introduction

When you set file parameters for a counter log, you must select a log file format. Select the log file format that is most appropriate for your environment. For example, if you are responsible for a few servers, the text file or the binary file format is best choice. If you are responsible for a hundred servers, logging your data to a SQL database is the best choice.

Procedure

To set file parameters for a counter log:

1. To start Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.
2. Double-click **Performance Logs and Alerts**.
3. To set file properties for a counter log, double-click **Counter Logs**.
4. In the details pane, double-click the log.
5. On the **Log Files** tab, complete the following options:
 - a. **Log file type**. In the list, select the format you want for this log file, complete the options, and then click the **Configure** button.
 - b. **Configure**. Select the configuration parameters using the following options for either **Configure Log Files** or **Configure SQL Logs**, based on the log file type that you selected in the **Log File type** list.
 - c. **End file names with**. Select this check box, and then, in the list, click the suffix style that you want to use. Use **End file names with** to distinguish between log files with the same log file name that are in a group of automatically generated logs.

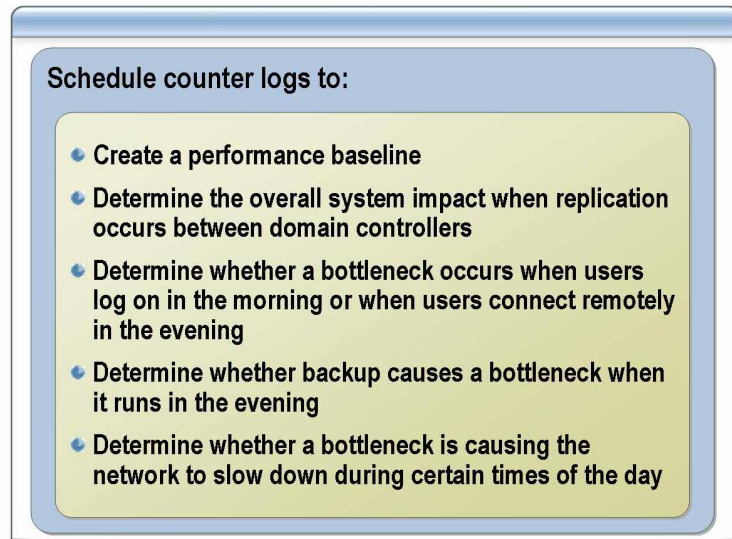
- d. **Start numbering at.** Set this option to the start number for automatic file numbering, when you select **nnnnnn** as the **End file names with**.
- e. **Comment.** If appropriate, type a comment or description for the log file.

Option	Description	Applies to
Location	Type the name of the folder in which you want to create the log file, or click Browse to search for the folder.	Configure Log Files
File name	Type a partial or base name for the log file. You can use File name in conjunction with End file names if appropriate.	Configure Log Files
Repository name	Select the System DSN (Data Source Name) from the drop-down list, and then type the Log set name. The Log set name will be stored in the database within the System DSN.	Configure SQL Logs

- 6. In the **Configure Log files** dialog box, under **Log file size**, use the following options:
 - a. **Maximum limit.** When you select this option, data is continuously collected in a log file until it reaches limits that are set by disk quotas or the operating system. For SQL logs, data is collected in a database until it reaches limits that are set by the number of records that are written.
 - b. **Limit of.** To define a size limit for the log file, specify the size. For counter and trace logs, specify the maximum size in megabytes. For SQL logs, specify the maximum size in records.

Note In the **Configure SQL Logs** dialog box, instead of **Log file size**, the option is called **Log set size**.

Why Schedule Counter Logs?



Introduction

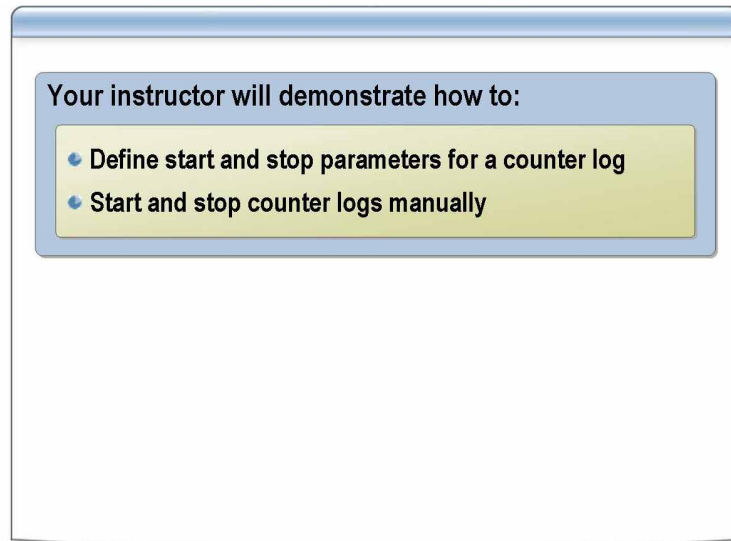
It is impractical for one person to monitor a network server 24 hours a day. You must automate this process so that you have time to perform your other tasks. You can schedule counter logs to create a performance baseline, look for bottlenecks, monitor system events, and collect information about how system events affect the server.

Why schedule counter logs?

You can schedule counter logs to:

- Create a performance baseline.
- Determine the effect on the overall system when replication occurs between domain controllers.
- Determine whether a bottleneck occurs when users log on in the morning.
- Determine whether a bottleneck occurs when users connect remotely in the evening.
- Determine whether Backup causes a bottleneck when it runs in the evening.
- Determine whether a bottleneck occurs during certain times of the day when users complain that the network slows down.

How to Schedule a Counter Log



Introduction

You typically schedule logging to occur during normal hours of operation. For most organizations, this period is between 8 A.M. and 5 P.M. For organizations that operate 24 hours a day and 7 days a week, logging should be turned on constantly. If logging is turned on constantly, you can create a log file for each shift (typically 8 hours), for the entire day (24 hours), or by size. A log file that is limited by size continues to grow to the size that you specify, and then a new log is started.

To schedule a counter log, you must define its start and stop parameters.

Procedure for defining start and stop parameters for a counter log

To define start and stop parameters for a counter log:

1. To start Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.
2. Double-click **Performance Logs and Alerts**, and then click **Counter Logs**.
3. In the details pane, double-click the name of the counter log.
4. On the **Schedule** tab, under **Start log**, click **At**, and then specify the time and date.
5. Under **Stop log**, select one of the following options:
 - a. To stop the log after a specified duration, click **After**, and then specify the number of intervals and the type of interval (days, hours, and so on).
 - b. To stop the log at a specific time and date, click **At**, and then specify the time and date. The year box accepts four characters; the others accept two characters.

- c. To stop a counter log when the log file becomes full, click **When the log file is full**. The file will continue to accumulate data according to the file-size limit that you set on the **Log Files** tab (in kilobytes up to two gigabytes).

Note Set the limit in the **Configure Log Files** dialog box before clicking **When the log file is full**. Otherwise, this option is deactivated.

Important When setting this option, take into consideration your available disk space and any disk quotas that are in place. An error can occur if your disk runs out of disk space due to logging.

6. Under **When a log file closes**, select the appropriate option:
 - a. If you want to configure a circular (continuous, automated) counter logging, select **Start a new log file**.
 - b. If you want to run a program after the log file stops, such as a copy command for transferring completed logs to an archive site), select **Run this command**. Also, type the path and file name of the program to run, or click **Browse** to locate the program.

Procedure for starting and stopping counter logs manually

In general, all logging should be automated and should follow a schedule. Logging should track usage of the servers during the period of greatest activity. There are times, however, when logging is not necessary, for example during periods of inactivity such as mandatory vacation times, holidays, system maintenance, and so on. During these times, you can stop logging manually.

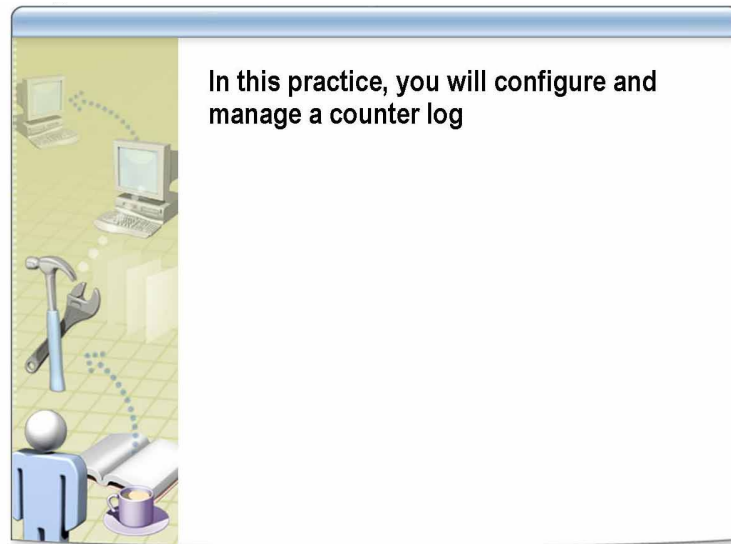
Likewise, logging may be needed outside of normal hours of operation, for example during periods of increased activity such as mandatory overtime during evenings and weekends. During these times, you can start logging manually.

To start and stop counter logs manually:

1. Open Performance, and then double-click **Performance Logs and Alerts**.
2. Click **Counter Logs**.
3. In the details pane, right-click the counter log that you want to stop or start.
4. Click **Start** or **Stop**.

Note You cannot check a counter log while it is running. You must stop a counter log to view it.

Practice: Configuring and Managing Counter Logs



Objective

In this practice, you will configure and manage a counter log.

Scenario

You are the systems administrator for a network. Your duties include monitoring servers at a remote data center. Recently, some users at your site complained about the speed of the Glasgow server in the data center. You monitor Glasgow by using System Monitor, but you do not detect any problems. To identify the problem, you decide to monitor Glasgow remotely by scheduling counter logs.

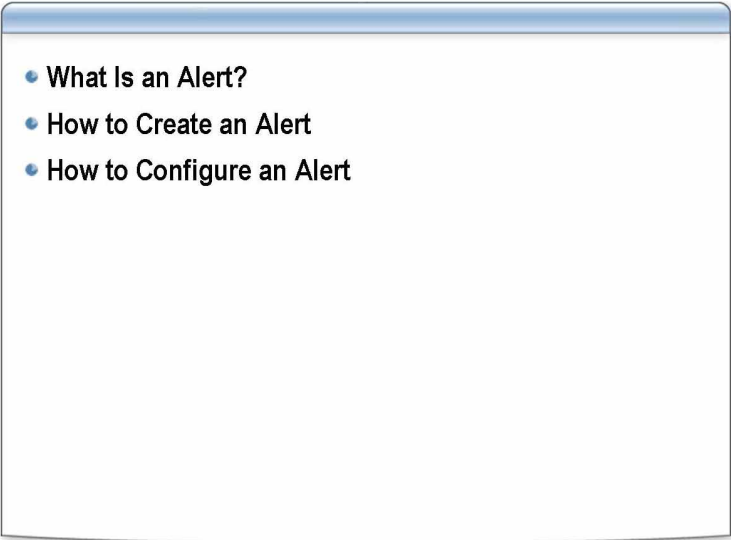
Practice

► Enable counter logs for a remote server

1. Log on to the domain as *ComputerUser* with a password of **P@ssw0rd**.
2. Open the **Run** dialog box and type **runas /user:nwtraders\Administrator "mmc %windir%\system32\perfmon.msc"** and then click **OK**.
3. When prompted for a password, type **P@ssw0rd** and press **ENTER**.
4. In the Performance console, expand **Performance Logs and Alerts**, open **Counter Logs**, and then create a new counter log named **Glasgow**.
5. In the **Glasgow** dialog box, open the **Add Counters** dialog box, and then in the **Select counters from computer** box, type **\\Glasgow**.
6. Add the following counters, and then close the **Add Counters** dialog box:
 - Processor\% Processor Time
 - Memory\Pages/sec
 - PhysicalDisk\Avg. Disk Queue Length
7. Set the interval to 1 second.
8. In the **Run as** box, type **nwtraders\Administrator** and then click **Set Password**.
9. In the **Password** and **Confirm Password** boxes, type **P@ssw0rd** and then click **OK**.

10. On the **Schedule** tab, set the **Start log** to begin two minutes from now.
11. Set the **Stop log** to stop three minutes from now, change the date to today's date, and then click **OK**.
12. If prompted to create a log folder, click **Yes**.
13. Wait until the time has elapsed.
14. Open the **Run** dialog box, type **runas /user:nwtraders\administrator cmd** and then press ENTER.
15. When prompted for a password, type **P@ssw0rd** and press ENTER.
16. Verify the existence of the performance log by typing the following command: **dir C:\Perflogs**
17. To view the log file, click **System Monitor**, and then right-click the graph and select **Properties**.
18. On the **Source** tab, select **Log files**, click **Add**, select the log file, and then click **Open**.
19. Click **Time Range**, adjust the time range for one minute, and then click **OK**.
20. Close all windows and log off.

Lesson: Configuring Alerts

- 
- What Is an Alert?
 - How to Create an Alert
 - How to Configure an Alert

Introduction

Use alerts to notify a user or an administrator when a predefined counter value exceeds or falls below a specified setting. In addition, you can use Performance Logs and Alerts to collect data about hardware resources, system services, and performance.

Lesson objectives

After completing this lesson, you will be able to:

- Explain an alert.
- Create an alert.
- Configure an alert.

What Is an Alert?

- **Feature that detects when a predefined counter value rises above or falls below a specified setting**
- **Specified setting on the counter is called alert threshold**
- **Set an alert on a counter when:**
 - Entry is made in application event log
 - Selected counter's value exceeds or falls below alert threshold
 - Message is sent
 - Program runs
- **Set alerts based on established performance baseline values**
- **Use alerts to be notified when a counter threshold value exceeds or falls below a specified value**

Definition

An *alert* is a feature that detects when a predefined counter value exceeds or falls below a specified setting. The specified setting on the counter is called the *alert threshold*.

Why use alerts?

By using the alert feature, you can define a counter value that triggers actions, such as sending a network message, running a program, or starting a log.

Alerts are useful if you are not actively monitoring a particular counter threshold value but want to be notified when it exceeds or falls below a specified setting so that you can investigate and determine the cause of the change. For example, you can set an alert to notify you when the number of failed logon attempts exceeds a specified number.

You may want to set alerts based on established performance baseline values for your system.

Functions of an alert

You can set an alert on a counter to perform the following functions:

- **Make an entry in the application event log.**

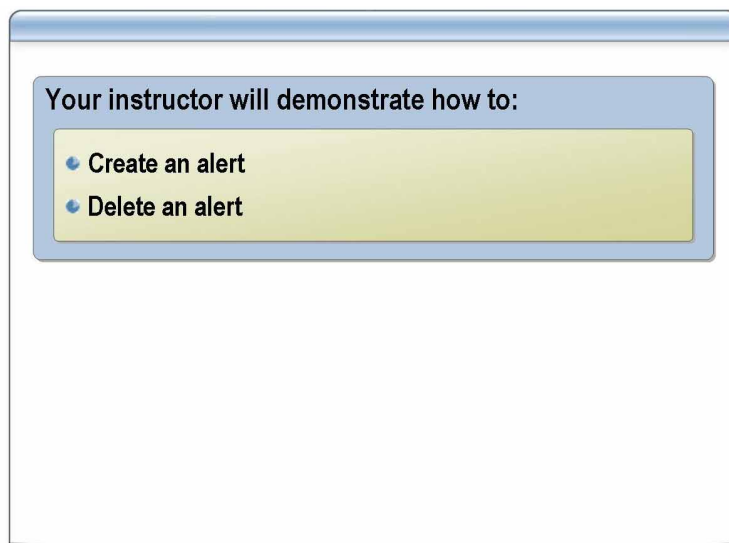
For example, enable this option if you want a record of all the events that cause an alert.
- **Start a log when the selected counter's value exceeds or falls below the alert threshold.**

For example, you can use this option to notify you if the processor usage time exceeds 85 percent.
- **Send a message.**

For example, enable this if you want to be alerted when a specific event occurs.
- **Run a program.**

Enable this option if you want a program to run when an event occurs. For example, you may want to shut down the server when the hard disk is full.

How to Create an Alert



Introduction

Use the following procedure to create an alert.

Procedure for creating an alert

To create an alert:

1. To open Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.
2. Double-click **Performance Logs and Alerts**, and then click **Alerts**.
3. Right-click a blank area of the details pane, and then click **New Alert Settings**.
4. In the **Name** box, type the name of the alert, and then click **OK**.

On the **General** tab, you can define a comment for your alert, along with counters, alert thresholds, and the sample interval.

On the **Action** tab, you can define the actions that occur when counter data triggers an alert.

On the **Schedule** tab, you can define when the service begins to scan for alerts.

Note To save the settings for an alert, right-click the alert in the right pane of the **Performance** console, and then click **Save Settings As**. You can then specify an .htm file in which to save the settings. To reuse the saved settings for a new alert, right-click the right pane, and then click **New Alert Settings From**. This is an easy way to generate new settings from an alert configuration. You can also open the HTML file in Internet Explorer to display a System Monitor graph.

Procedure for deleting an alert

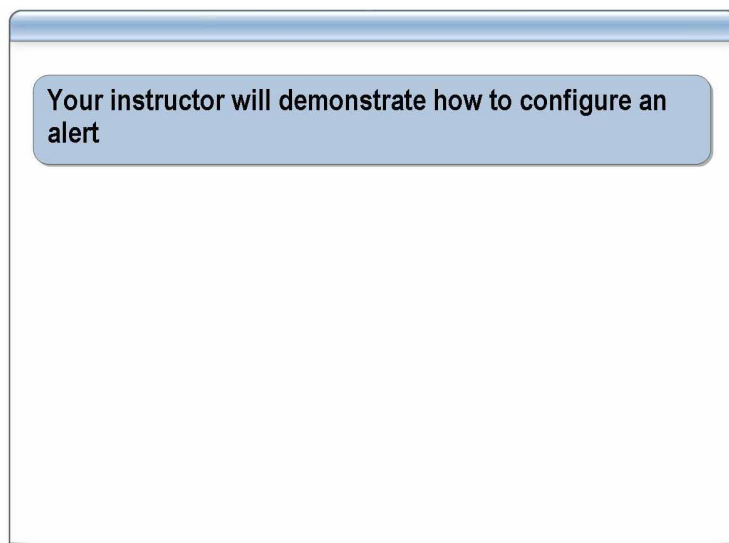
To delete an alert:

1. To start Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.
2. Double-click **Performance Logs and Alerts**.
3. In the details pane, right-click the alert that you want to delete.
4. Click **Delete**.

Important To perform the preceding procedures, you must be a member of the Administrators group, or you must have been delegated the appropriate authority. If the computer is connected to a domain, members of the **Domain Admins** group might be able to perform this procedure.

As a security best practice, consider using **Run as** to perform this procedure.

How to Configure an Alert



Introduction

Use the following procedure to configure an alert.

Procedure

1. To open Performance, click **Start**, point to **Administrative Tools**, and then click **Performance**.
2. Double-click **Performance Logs and Alerts**, and then click **Alerts**.
3. In the details pane, double-click the alert.
4. On the **General** tab, in the **Comment** box, type a comment to describe the alert as needed, and then click **Add**.
5. For each counter or group of counters that you want to add to the log, perform the following steps:
 - a. To monitor counters from the computer on which Performance Logs and Alerts will run, click **Use local computer counters**.

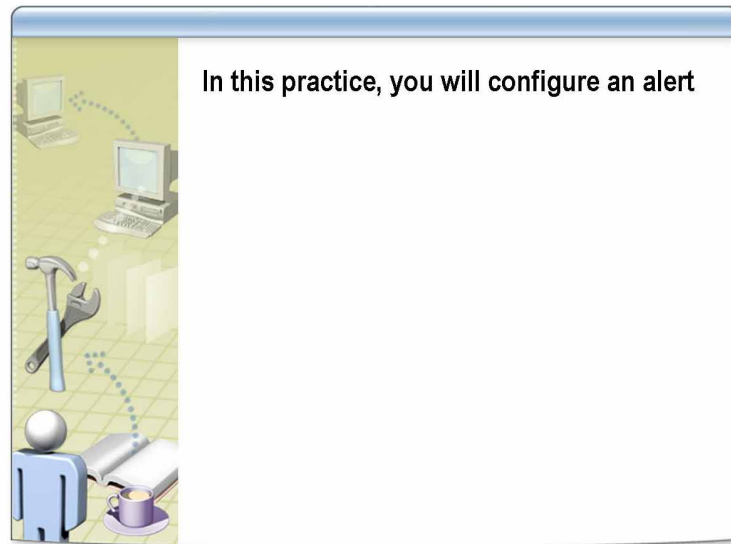
Or, to monitor counters from a specific computer regardless of where the service is run, click **Select counters from computer**, and then type the name of the computer that you want to monitor.
 - b. Under **Performance object**, click a performance object to monitor.
 - c. Under **Select counters from list**, click one or more counters to monitor.
 - d. To monitor all instances of the selected counters, click **All Instances**.
Binary logs can include instances that are not available at log startup but subsequently become available.

Or, to monitor particular instances of the selected counters, click **Select instances from list**, and then click an instance or instances to monitor.
 - e. Click **Add**, and then click **Close**.
6. Under **Alert when the value is**, specify **Under** or **Over**, and in **Limit**, specify the value that triggers the alert. Perform this step for each counter or group of counters that you added to the log.

In **Sample data every**, specify the amount and the unit of measure for the update interval.

7. On the **Schedule** tab, under **Start Scan**, click one of the following options:
 - To start alert manually, click **Manually**. When this option is selected, to start the log or alert, right-click the log name in the details pane, and then click **Start**.
 - To start alert at a specific time and date, click **At**, and then specify the time and date.
 - Under **Stop Scan**, select one of the following options:
 - To stop the alert manually, click **Manually**. When this option is selected, to stop the log or alert, right-click the log or alert name in the details pane, and then click **Stop**.
 - To stop the alert after a specified duration, click **After**, and then specify the number of intervals and the type of interval (days, hours, and so on).
 - To stop the alert at a specific time and date, click **At**, and then specify the time and date. (The year box accepts four characters; the others accept two characters.)
8. Under **When an alert scan finishes**, select **Start a new scan** if you want to configure continuous alert scanning.

Practice: Configuring an Alert



Objective

In this practice, you will configure an alert.

Scenario

You are the systems administrator for an organizational unit on a network. The organizational unit recently acquired a new file server. You determine that the processor on this server should never exceed 50 percent usage. You want to configure an alert to warn you when the processor exceeds this figure, and you also must test the alert.

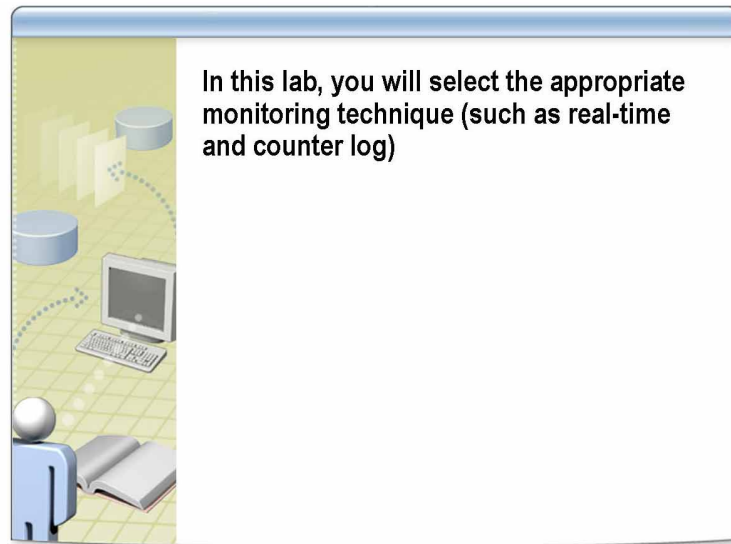
Practice

► **Configure an alert**

1. Log on as *ComputerUser* with a password of **P@ssw0rd**.
2. Open the **Run** dialog box, type **runas /env /user:nwtraders\administrator cmd** and then click **OK**.
3. When prompted for a password, type **P@ssw0rd** and then press ENTER.
4. In the Command Prompt window, start **compmgmt.msc**.
5. In the Command Prompt window, start **perfmon.msc**.
6. In the Performance console, open **Alerts** and then create a new alert named **CPU Alert**.
7. Add the Processor\% Processor Time counter, and then close the **Add Counters** dialog box.
8. Change the **Limit** to **50** and then close the **CPU Alert** dialog box.
9. In the Cmd window, change to the **C:\MOC\2275\Labfiles** folder.
10. In the command line, run the **CPULoop** batch file.

11. Stop **CPULoop** by closing the Cmd window.
12. In Computer Management, expand **System Tools**, expand **Event Viewer**, and then open **Application**.
13. Open the first entry and in the **Event Properties** dialog box, read the entry under **Description**.
14. Close all windows and log off.

Lab A: Preparing to Monitor Server Performance



Objectives

After completing this lab, you will be able to examine various scenarios and select the appropriate monitoring technique.

Prerequisites

None.

**Estimated time to
complete this lab:**
20 minutes

Exercise 1

Selecting the Appropriate Monitoring Technique

In this exercise, you will select the appropriate monitoring technique based on the following scenarios. R=Real Time, L=Logging, A=Alerts. If more than one technique will work, put your selections in order of preference.

Scenario	Monitoring technique(s)
1. Determine when the hard disk is running out of free space.	
2. Provide management with information that can be used for budgeting purposes.	
3. Determine the number of users that a specific server configuration should support.	
4. Analyze a trend.	
5. Monitor multiple servers.	
6. Determine when to increase capacity.	
7. Find intermittent performance problems.	
8. Investigate why a computer application is slow or inefficient.	
9. Determine when to add additional system resources.	
10. Determine when to upgrade the system.	
11. Determine how a server should be used.	
12. Determine expected response times for specific numbers of users and system use.	
13. Analyze data to find and resolve abnormalities in the system use.	
14. Monitor use over time.	
15. Determine a preventive maintenance schedule for your servers.	
16. Create a baseline for a server.	
17. Monitor the effects of replication.	
18. Troubleshoot a server.	
19. Plan for growth.	
20. Find a slow memory leak.	

(continued)

Scenario	Monitoring technique(s)
21. Find a fast memory leak.	
22. Monitor intermittent disk thrashing.	
23. Monitor continuous disk thrashing	
24. Monitor a remote computer.	
25. Respond to user complaints that a server seems to be running slowly.	
26. Monitor a computer 24 hours a day, 7 days a week.	