**Table 5.5 Comparison of Process Data Supplied by Task Manager and System Monitor** *(continued)*

| Task Manager process measure | Description | System Monitor process object counters |
|---|---|---|
| I/O Write Bytes | The number of bytes written in input/output operations generated by a process, including file, network, and device I/Os. I/O Write Bytes directed to CONSOLE (console input object) handles are not counted. | I/O Write Bytes/sec |
| I/O Writes | The number of write input/output operations generated by a process, including file, network, and device I/Os. I/O Writes directed to CONSOLE (console input object) handles are not counted. | I/O Write Operations/sec |
| Image Name | Name of the process. | The process name in the **Instances** box |
| Memory Usage | The amount of main memory, in kilobytes, used by the process. | Working Set |
| Memory Usage Delta | The change in memory use, in kilobytes, since the last update. Unlike System Monitor, Task Manager displays negative values. | None |
| Nonpaged Pool | The amount of memory, in kilobytes, used by a process. Operating system memory that is never paged to disk. Paging is the moving of infrequently used parts of a program's working memory from RAM to another storage medium, usually the hard disk. | Pool Nonpaged Bytes |
| Page Faults | The number of times that data had to be retrieved from disk for this process because it was not found in memory. This value is accumulated from the time the process is started. | None<br>Page faults/sec is the rate of page faults over time. |
| Page Faults Delta | The change in the number of page faults since the last update. | None |
| Paged Pool | The amount of system-allocated virtual memory, in kilobytes, used by a process. The paged pool is virtual memory available to be paged to disk. Paging is the moving of infrequently used parts of a program's working memory from RAM to another storage medium, usually the hard disk. The paged pool includes all of user memory and a portion of system memory. | Pool Paged Bytes |
| Peak Memory Usage | The peak amount of physical memory resident in a process since it started. | None |

*(continued)*

**Table 5.5 Comparison of Process Data Supplied by Task Manager and System Monitor** ***(continued)***

| Task Manager process measure | Description | System Monitor process object counters |
|---|---|---|
| PID (Process Identifier) | Numerical ID assigned to the process while it runs. | ID Process |
| Thread Count | The number of threads running in the process. | Thread Count |
| USER Objects | The number of USER objects currently being used by a process. A USER object is an object from Window Manager, which includes windows, menus, cursors, icons, hooks, accelerators, monitors, keyboard layouts, and other internal objects. | None |
| Virtual Memory Size | The amount of virtual memory, or address space, committed to a process. | Private Bytes |

For more information about Task Manager and its use in monitoring processor and memory performance, see the following chapters in this book:

- "Evaluating Memory and Cache Usage"
- "Analyzing Processor Activity"
- "Measuring Multiprocessor System Activity"

## Monitoring the System

To see a dynamic overview of system performance, including a graph and numeric display of processor and memory usage, click the Task Manager **Performance** tab, as shown in Figure 5.11.

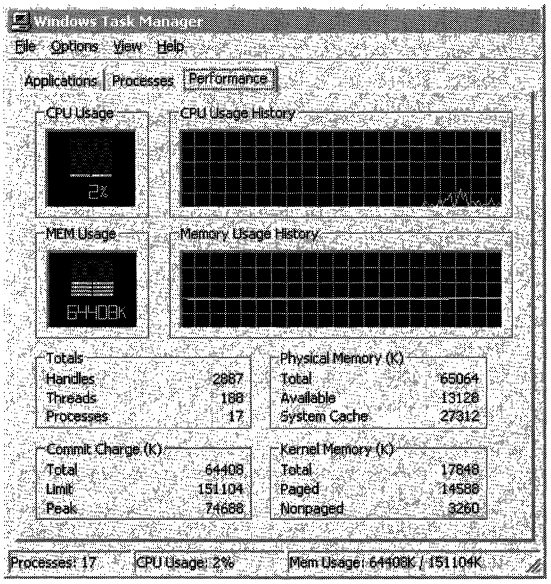



**Figure 5.11 Task Manager Performance Tab**

To graph the percentage of processor time in privileged or kernel mode, click **Show Kernel Times** on the **View** menu. This is a measure of the time that applications are using operating system services. The remaining time, known as user mode, is spent running threads within the application code.

Users of multiple-processor computers can click **CPU History** on the **View** menu and then graph the nonidle time of each processor in a single graph or in separate graphs.

The following table briefly describes the counts on the Performance tab and their System Monitor counterparts, if any.

**Table 5.6 Comparison of System Data Provided by Task Manager and System Monitor**

| Task Manager counts | Description | System Monitor counters |
|---|---|---|
| CPU Usage | The percentage of time the processor is running a thread other than the Idle thread. | Processor\% Processor Time |
| MEM Usage | The amount of virtual memory used, in kilobytes. | Memory\Committed Bytes |
| Total Handles | The number of object handles in the tables of all processes. | Process(_Total)\Handle Count |

*(continued)*

**Table 5.6 Comparison of System Data Provided by Task Manager and System Monitor *(continued)***

| Task Manager counts | Description | System Monitor counters |
|---|---|---|
| Total Threads | The number of running threads, including one Idle thread per processor. | Process(_Total)\Thread Count |
| Total Processes | The number of active processes, including the Idle process. | Object\Processes is the same, but excludes the Idle process. |
| Physical Memory: Total | Amount of physical, random access memory, in kilobytes, installed in the computer. | None |
| Physical Memory: Available | Amount of physical memory available to processes, in kilobytes.<br>It includes zeroed, free, and standby memory. | Memory\Available Bytes |
| Physical Memory: File Cache | Amount of physical memory, in kilobytes, released to the file cache on demand. | Memory\Cache Bytes |
| Commit Charge: Total | Size of virtual memory in use by all processes, in kilobytes. | Memory\Committed Bytes |
| Commit Charge: Limit | Amount of virtual memory, in kilobytes, that can be committed to all processes without enlarging the paging file. | Memory\Commit Limit |
| Commit Charge: Peak | The maximum amount of virtual memory, in kilobytes, used in the session.<br>The commit peak can exceed the commit limit if virtual memory is expanded. | None |
| Kernel Memory: Total | Sum of paged and nonpaged memory, in kilobytes. | None<br>(Sum of Pool Paged Bytes and Pool Nonpaged Bytes) |
| Kernel Memory: Paged | Size of the paged pool, in kilobytes, allocated to the operating system. | Memory\Pool Paged Bytes |
| Kernel Memory: Nonpaged | Size of the nonpaged pool, in kilobytes, allocated to the operating system. | Memory\Pool Nonpaged Bytes |

## Resource Kit Performance Tools

The *Windows 2000 Resource Kit* companion CD contains other performance-related tools, such as the following:

- **Ctrlist.exe.** This prints counter descriptions to a file or to the screen.
- **Extctrlst.exe.** This lists all counter DLLs that are running and provides the capability of disabling them. For more information about disabling counters, see "Troubleshooting Problems with Performance Tools" later in this chapter.
- **Perfmtr.exe.** This command-line utility is useful for dynamically monitoring performance statistics relating to memory, processor, and I/O activity.
- **Showperf.exe.** This program is useful for developers who want to see the counter type, index, and the contents of the Performance Data block so they can view and debug the counter's raw data structure.
- **Typeperf.exe.** This is a command-line utility for displaying performance information from individual performance counters.

For information about the utilities listed, see the *Windows 2000 Resource Kit* Tools Help. Programs used in specific bottleneck analysis contexts are described in the chapters to which they apply.

# Starting Your Monitoring Routine

Setting up a monitoring routine consists of several steps, including setting up a basic monitoring configuration (sometimes called "overview" settings), testing the limits of acceptable performance under various conditions, and establishing a baseline. The following sections describe how to undertake these steps.

## Your Minimum Monitoring Configuration

The minimum performance objects to monitor are those corresponding to the main hardware resources of your system: memory, processors, disks, and network components. Table 5.7 lists the appropriate counters and the categories of information they provide.

**Table 5.7 Monitoring the Minimum Objects**

| Component | Performance aspect being monitored | Counters to monitor |
|---|---|---|
| Disk | Usage | LogicalDisk\% Free Space |
| | | LogicalDisk\% Disk Time |
| | | PhysicalDisk\Disk Reads/sec |
| | | PhysicalDisk\Disk Writes/sec |
| | | Use **diskperf -y** to enable disk counters and **diskperf -n** to disable them. To specify the type of counters you want to activate, include **d** for physical disk drives and **v** for logical disk drives or storage volumes. When the operating system starts up, it automatically sets the **diskperf** command with the **-yd** switch to activate physical disk counters. Type **diskperf -yv** to activate logical disk counters. For more information about using the **diskperf** command, type **diskperf -?** at the command prompt. |
| | | The % Disk Time counter must be interpreted carefully. Because the _Total instance of this counter might not accurately reflect utilization on multiple-disk systems, it is important to use the % Idle Time counter as well. Note that these counters cannot display a value exceeding 100 percent. |
| | | For more information about disk performance counters, see "Examining and Tuning Disk Performance" in this book. |
| Disk | Bottlenecks | LogicalDisk\Avg. Disk Queue Length |
| | | PhysicalDisk\Avg. Disk Queue Length (all instances) |
| Memory | Usage | Memory\Available Bytes |
| | | Memory\Cache Bytes |
| | | You can also use Memory\Committed Bytes and Memory\Commit Limit to detect problems with virtual memory. |

*(continued)*

**Table 5.7 Monitoring the Minimum Objects *(continued)***

| Component | Performance aspect being monitored | Counters to monitor |
|---|---|---|
| Memory | Bottlenecks or leaks | Memory\Pages/sec |
| | | Memory\Page Faults/sec |
| | | Memory\Pages Input/sec |
| | | Memory\Page Reads/sec |
| | | Memory\Transition Faults/sec |
| | | Memory\Pool Paged Bytes |
| | | Memory\Pool Nonpaged Bytes |
| | | Although not specifically Memory object counters, the following are also useful for memory analysis: |
| | | Paging File\% Usage Object (all instances) |
| | | Cache\Data Map Hits % |
| | | Server\Pool Paged Bytes and Server\Pool Nonpaged Bytes |
| Network | Usage | Network Segment: % Net Utilization |
| | | Note that you need to install the Network Packet Protocol driver for Network Monitor in order to use this counter. |
| Network | Throughput | Protocol transmission counters (varies with networking protocol); for TCP/IP: |
| | | Network Interface\Bytes total/sec |
| | | Network Interface\Packets/sec |
| | | Server\Bytes Total/sec or Server\Bytes Sent/sec and Server\Bytes Received/sec |
| | | You might want to monitor other objects for network and server throughput, as described in "Monitoring Network Performance" in this book.. |
| Processor | Usage | Processor\% Processor Time (all instances) |
| | Bottlenecks | System\Processor Queue Length (all instances) |
| | | Processor\Interrupts/sec |
| | | System\Context switches/sec |

If you want to test the limits of your system as part of establishing a baseline, monitor the recommended counters during the following activities:

- Adding base services
- Adding connections
- Running network applications
- Opening a file

Exhibit 1012
Page 291 of 776

- Printing a file
- Copying or writing to a file
- Accessing a database
- Sending a message

## Establishing the Baseline

After becoming familiar with System Monitor and the process of configuring graphs and logs, you are ready to incorporate monitoring into your daily routine of system administration. Routine monitoring over periods ranging from days to weeks to months allows you to establish a baseline for system performance.

A baseline is a measurement that is derived from the 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.

When determining your baseline, it is very important to know the types of work being done and the days and times when the work is being done. That will help you to associate work with resource usage and to determine the reasonableness of performance during those intervals.

For example, if you find that performance diminishes somewhat for a brief period at a given time of day, and you find that at that time many users are logging on or off, it might be an acceptable slowdown. Similarly, if you find that performance is poor every evening at a certain time and you can tell that that time coincides with nightly backups when no users are logged on to the system, again that performance loss might be acceptable. But you can make that determination only when you know the degree of performance loss and its cause.

When you have built up data on performance over a period, with data reflecting periods of low, average, and peak usage, you can make a subjective determination of what constitutes acceptable performance for your system. That determination is your baseline. Use your baseline to detect when bottlenecks are developing or to watch for long-term changes in usage patterns that require you to increase capacity.

# Analyzing Monitoring Results

The baseline you develop establishes the typical counter values you should expect to see when your system is performing satisfactorily. The following section provides guidelines to help you interpret the counter values and eliminate false or misleading data that might cause you to set your own target values inappropriately.

When you are collecting and evaluating data to establish a valid performance baseline, consider the following guidelines:

- When monitoring processes of the same name, watch for unusually large values for one instance and not the other. This can occur because System Monitor sometimes misrepresents data for separate instances of processes of the same name by reporting the combined values of the instances as the value of a single instance. Tracking processes by process identifier can help you get around this problem. For information about monitoring processes, see "Analyzing Processor Activity" later in this book.
- When you are monitoring several threads and one of them stops, the data for one thread might appear to be reported for another. This is because of the way threads are numbered. For example, you begin monitoring and have three threads, numbered 0, 1, and 2. If one of them stops, all remaining threads are resequenced. That means that the original thread 0 is now gone and the original thread 1 is renamed to 0. As a result, data for the stopped thread 0 could be reported along with data for the running thread 1 because old thread 1 is now old thread 0. To get around this problem, you can include the thread identifiers of the process's threads in your log or display. Use the Thread\Thread ID counter for this purpose.
- Do not give too much weight to occasional spikes in data. These might be due to startup of a process and are not an accurate reflection of counter values for that process over time. The effect of spikes can linger over time when using counters that average.
- For monitoring over an extended period of time, use graphs instead of reports or histograms because these views only show last values and averages. As a result, they might not give an accurate picture of values if you are looking for spikes.
- Unless you specifically want to include startup events in your baseline, exclude these events because the temporary high values tend to skew overall performance results.
- Investigate zero values or missing data. These can impede your ability to establish a meaningful baseline. There are several possible explanations for this. For more information, see "Troubleshooting Problems with Performance Tools" later in this chapter.

# Identifying Potential Bottlenecks

Deviations from your baseline provide the best indicator of performance problems. However, as a secondary reference, the following table describes recommended thresholds for object counters. Use this table to help you identify when a performance problem is developing on your system. If the values listed are consistently reported on your system, consult additional chapters in the Performance Monitoring section of this book for how to investigate and correct the problems causing these values.

**Table 5.8 Recommended Thresholds for the Minimum Set of System Counters**

| Resource | Object\Counter | Suggested threshold | Comments |
|---|---|---|---|
| Disk | LogicalDisk\% Free Space | 15 percent | None |
| Disk | LogicalDisk\% Disk Time | 90 percent | None |
| Disk | PhysicalDisk\ Disk Reads/sec, PhysicalDisk\ Disk Writes/sec | Depends on manufacturer's specifications | Check the specified transfer rate for your disks to verify that this rate does not exceed the specifications. In general, Ultra Wide SCSI disks can handle 50 to 70 I/O operations per second. |
| Disk | PhysicalDisk\ Current Disk Queue Length | Number of spindles plus 2 | This is an instantaneous counter; observe its value over several intervals. For an average over time, use PhysicalDisk\Avg. Disk Queue Length. |
| Memory | Memory\ Available Bytes | Less than 4 MB | Research memory usage and add memory if needed. |
| Memory | Memory\ Pages/sec | 20 | Research paging activity. |
| Network | Network Segment\% Net Utilization | Depends on type of network | You must determine the threshold based on the type of network you are running. For Ethernet networks, for example, 30 percent is the recommended threshold. |
| Paging File | Paging File\% Usage | Above 70 percent | Review this value in conjunction with Available Bytes and Pages/sec to understand paging activity on your computer. |
| Processor | Processor\% Processor Time | 85 percent | Find the process that is using a high percentage of processor time. Upgrade to a faster processor or install an additional processor. |

*(continued)*

**Table 5.8 Recommended Thresholds for the Minimum Set of System Counters *(continued)***

| Resource | Object\Counter | Suggested threshold | Comments |
|---|---|---|---|
| Processor | Processor\ Interrupts/sec | Depends on processor; for current CPUs, use a threshold of 1500 interrupts per second. | A dramatic increase in this counter value without a corresponding increase in system activity indicates a hardware problem. Identify the network adapter or disk controller card causing the interrupts. You might need to install an additional adapter or controller card. |
| Server | Server\Bytes Total/sec | | If the sum of Bytes Total/sec for all servers is roughly equal to the maximum transfer rates of your network, you might need to segment the network. |
| Server | Server\Work Item Shortages | 3 | If the value reaches this threshold, consider tuning the InitWorkItems or MaxWorkItems entries in the registry (in HKEY_LOCAL_MACHINE \SYSTEM\CurrentControlSet \Services\lanmanserver\Parameters). For more information about MaxWorkItems, see the Microsoft Knowledge Base link on the Web Resources page at http://windows.microsoft.com /windows2000/reskit/webresources<br><br>**Caution** Do not use a registry editor to edit the registry directly unless you have no alternative. The registry editors bypass the standard safeguards provided by administrative tools. These safeguards prevent you from entering conflicting settings or settings that are likely to degrade performance or damage your system. Editing the registry directly can have serious, unexpected consequences that can prevent the system from starting and require that you reinstall Windows 2000. To configure or customize Windows 2000, use the programs in Control Panel or Microsoft Management Console whenever possible. |
| Server | Server Work Queues\Queue Length | 4 | If the value reaches this threshold, there might be a processor bottleneck. This is an instantaneous counter; observe its value over several intervals. |
| Multiple Processors | System\Processor Queue Length | 2 | This is an instantaneous counter; observe its value over several intervals. |

# Investigating Bottlenecks

Investigating performance problems should always start with monitoring the whole system before looking at individual components. In precise terms, a bottleneck exists if a particular component's limitation is keeping the entire system from performing more quickly. Therefore, even if one or more components in your system is heavily used, if other components or the system as a whole show no adverse effects, then there is no bottleneck.

For example, suppose that a process had 10 threads, each of which used exactly 0.999 seconds of processor time once every 10 seconds. If each thread made a request exactly 1 second after the previous one in perfect sequence, the processor would be 99.9 percent busy, but there would be no queue, no interference between the threads, and, technically, no bottleneck, although the system probably could not support any increased load or variation in its request scheduling without creating one.

Factors involved in the development of a bottleneck are the number of requests for service, the frequency with which requests occur, and the duration of each request. As long as these are perfectly synchronized, no queue will develop and no bottleneck will arise. The device with the smallest throughput ratio is probably the primary source of the bottleneck.

It is difficult to detect multiple bottlenecks in a system. You might spend several days testing and retesting to identify and eliminate a bottleneck, only to find that another appears in its place. Only thorough and patient testing of all elements can ensure that you have found all of the problems.

It is not unusual to trace a performance problem to multiple sources. Poor response time on a workstation is most likely to result from memory and processor problems. Servers are more susceptible to disk and network problems.

Also, problems in one component might be the *result* of problems in another component, not the cause. For example, when memory is scarce, the system begins moving pages of code and data between disks and physical memory. The memory shortage becomes evident from increased disk and processor use, but the problem is memory, not the processor or disk.

If you identify a resource that is out of range for your baseline or based on the recommended thresholds discussed in the preceding section, you need to investigate the activity of that resource in greater detail. This includes the following steps:

- Analyze your hardware and software configurations. Does your configuration match Microsoft recommendations for the operating system and the services you are supporting?

- Review entries in the event log for the time period when you begin seeing out-of-range counter values; these entries might provide information on problems that might result in poor system performance.
- Examine the kinds of applications you are running and what resources they demand, to determine their adequacy.
- Consider variables in your workload, such as processing different jobs at different times. For more efficient analysis, when you are looking for a specific problem, limit your charts and reports to specific events occurring at known times.
- For immediate diagnosis and problem solving of situations such as shutdowns and logon failures, log or monitor for a shorter time. Sampling should be frequent when monitoring over a short period. Similarly, for long-term planning and analysis, log for a longer period and set the update interval accordingly.
- Consider network or disk utilization or other activities occurring at the times that you see increasing resource utilization. Try to understand the usage patterns. Are they associated with specific protocols or computers?
- Approach bottleneck correction in a scientific manner. For example, never make more than one change at a time, always repeat monitoring after a change to validate the results, eliminate results that are suspect, and keep good records of what you have done and what you have learned.

When investigating bottlenecks in specific resources, focus on the performance objects and counters that pertain to the specific resource that appears to be your bottleneck. Your reference for information about these counters and how to detect and correct bottlenecks should be the chapter of this guide that refers to the resource you are investigating. These chapters also discuss how to use other Windows 2000 tools and utilities on the *Windows 2000 Resource Kit* companion CD for bottleneck detection and tuning. The chapters are as follows:

- "Evaluating Memory and Cache Usage"
- "Analyzing Processor Activity"
- "Examining and Tuning Disk Performance"
- "Monitoring Network Performance"
- "Measuring Multiprocessor System Activity"

# Troubleshooting Problems with Performance Tools

Occasionally you might have problems obtaining performance data, or you might find that monitoring a process is adding an unnecessary load to a computer you are monitoring. The following sections discuss how to handle some of these problems:

- Investigating Zero Values
- Investigating Other Problems with Tool Usage
- Controlling Performance Monitoring Overhead

## Investigating Zero Values

If data for selected counters is consistently reported as zeroes, this might indicate a problem with the counters or the way you are using the performance tools, rather than just the absence of any nonzero data. Following are descriptions of the possible causes and solutions to problems that result in missing or zero counter values.

- The process being monitored has stopped and, as a result, there is no data for the process in the performance tools. If you stopped the process manually, restart it to see the process in System Monitor. Otherwise, check Event Viewer for concurrent entries. You might find an error associated with this process.
- The counter DLL was disabled after you selected the corresponding counters in a log or display. The performance tools will not detect that the counter was removed or disabled, but will report the counter data as zeroes.
- You are attempting to monitor a computer that you don't have permission to access. This might occur if you are using a saved console that specifies a particular computer name. This causes System Monitor to report data as zeroes. It also causes System Monitor to start up slowly. For more information see "Security Issues" later in this chapter.

## Investigating Other Issues with Performance Tools

You might occasionally believe that data reported by the performance tools is invalid, that data is incomplete, or that the tools are not operating properly. This section addresses some problems you might encounter with the tools and how to correct them.

### Data seems to be missing.

- System Monitor might show gaps in its line graphs because data collection was subordinated to higher-priority processing activity on a system with a heavy load. When the system has adequate resources to continue with data collection, the graphing will resume as usual. A message appears describing this. Note also that you might see delays in display of data for some counters. Counters that display an average must wait for two samples to elapse before displaying a value. For ways to reduce the performance overhead of system monitoring, see "Controlling Performance Monitoring Overhead" later in this chapter.
- Values recorded in a log do not appear in the graph view. This is because the graph is limited to 100 samples. Reducing the size of the Time Window on the **Data** property tab can allow you to see a more complete range of data.

### Objects, counters, or instances seem to be missing or invalid.

In some cases you might find that you cannot find an object you want to monitor. This might be caused by the following:

- You have not started or installed the process that starts the object counters. Use Task Manager to verify that the process is running. If so, use Exctrlst.exe on the *Windows 2000 Resource Kit* companion CD to verify that the counter DLL is running.
- You have not enabled the counters (such as with the Network Segment object counters). If you do not see a counter that you want to monitor, make sure that the service or feature that provides the counter has been installed or configured. For information about how to install or configure the service or feature, see online Help for the service or feature.
- If counters have been disabled, then they will not appear in the Add Counters dialog box. There are several reasons that a counter DLL might be disabled:
  - A user disables counter DLLs using Exctrlst.exe on the *Windows 2000 Resource Kit* companion CD.
  - The Performance Library's built-in testing routines have found problems with the counter DLLs and have disabled them to prevent them from interfering with operation of the Performance console. If this has occurred, the Application Log in Event Viewer contains a message to this effect.

To re-enable the counters for debugging purposes, locate the Performance subkey under the subkey for the service (typically, HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services *service_name*\Performance\) and change the value for the Disable Performance Counters entry from 1 (disabled) to 0 (enabled). Notice that a counter DLL that the system has automatically disabled is likely to contain errors and may cause the system to slow.

- You lack permissions on the computer being monitored. If you do not have appropriate permissions to monitor the computer, an error message will be displayed when you attempt to select the computer. An administrator must ensure that your user account has permissions to use the performance tools. If you are trying to monitor a remote system, for information about security issues, see "Monitoring Remote Computers" later in this chapter.
- The DLL that installs the counters is generating errors. An example is if the counter does not handle localization functions correctly. Check Event Viewer to see if the counter DLL or the Performance Data Helper (PDH) library reported any errors. If necessary, you can disable counter DLLs that are causing errors by using Exctrlst.exe on the *Windows 2000 Resource Kit* companion CD.
- You are trying to monitor a 16-bit or MS-DOS application. Only 32-bit processes appear in the **Instances** box. Active 16-bit processes appear as threads running in a Windows NT Virtual DOS Machine (NTVDM) process. (Virtual DOS Machine is an environment system for MS-DOS and 16-bit Windows emulation.) If you want to monitor a 16-bit application, see "Monitoring 16-bit Windows Applications" later in this chapter.

---

**Note** When trying to monitor administrative tools hosted in MMC, note that they appear as instances of MMC in the **Add Counters** dialog box.

---

- The instance you want to monitor is not currently active. If you are configuring System Monitor to collect real-time data, you can only select active instances for data collection. (If you are viewing logged data, you can select inactive instances for which the log contains data.) If you select the process and it stops after you have selected it, it will continue to appear in the list box but the reported data will be zeroes.
- You might see situations where an instance seems inappropriate for the counter—such as the _Total instance for the Process\ID Process counter. All counters for an object have the same instances.

### Data seems invalid.

There are several reasons that counters might report unlikely values:

- You sometimes see an extremely high value for one instance and not the other when you are monitoring processes of the same name. This is because the performance tools sometimes misrepresent data for separate instances of processes with the same name by reporting the combined values of the instances as the value of a single instance. Using the instance index and tracking the Process\ID Process and Process\Creating Process ID counters can help you get around this problem.
- Also, when monitoring several threads and one of them stops, the data for one thread might appear to be reported for another. This is because of the way threads are numbered. For example, you begin monitoring and have three threads, numbered 0, 1, and 2. If one of them stops, all remaining threads are resequenced. That means that the original thread 0 is now gone and the original thread 1 is renamed to 0. As a result, data for the stopped thread 0 could be reported along with data for the running thread 1 because old thread 1 is now old thread 0. Again, using the instance index can help you to track these threads.

### Problems with System Monitor and MMC

- If you are trying to create a custom console with System Monitor and another tool, you might have a problem because System Monitor is not listed in the **Add Standalone Snap-in** dialog box. This is because System Monitor is not designed as an extension snap-in but as an ActiveX control. To create a custom console containing System Monitor, you select **ActiveX Control** in the **Add Standalone Snap-in** dialog box and select **System Monitor Control** in the **Insert ActiveX Control** dialog box. System Monitor Control will appear as the name of the utility in your custom console; you can change it as needed.
- Help for System Monitor does not appear in MMC; only Performance Logs and Alerts is shown in MMC Help. Because System Monitor is designed as an ActiveX control, it is unlike other MMC snap-ins. For example, System Monitor Help is not available by clicking **Help Topics** on the **Help** menu or by right-clicking **System Monitor** and selecting **Help** in the shortcut menu. Instead, click **Help** on the System Monitor display toolbar.

### Problems with logs

- Trace log data is not output as readable text. You must use a parsing utility to interpret the trace log output. Developers can create such a utility with the APIs provided in the Platform Software Development Kit.

- An error message appears if you try to export log data to Microsoft Excel while the Performance Logs and Alerts service is actively collecting data to that log. The service must be stopped because Excel requires exclusive access to the log file. Other programs are not known to require this exclusive access; therefore, in general, you can work with data from a log file while the service is collecting data to that file.
- Data from a running log does not seem to be updating. If you are working in System Monitor with a log file that is currently collecting data, you will need to click **Select Time Range** and keep moving the **Time Range** bar to the right to update the display with new samples.
- Processes that started while a log was running do not appear in my exported log. Logged data can be saved as comma-separated or tab-separated files that are easily viewed with Microsoft Excel. However, some limitations apply when you use this format. Instances that start after the log is started will not be reflected. You need to use the binary log format to see data for these instances.
- Errors occur regarding counter log size. This could be because your counter log has consumed the available space on the hard drive that you specified in the log file path. In addition, counter logs have a maximum size of 2 GB, and a message will appear when the log reaches this limit. Note that logs this size are unwieldy and slow to work with.
- An exported monitoring configuration is collecting data from the wrong computer. This is probably because you selected the **Select counters from computer** option when you saved the console and installed it on another system. Instead, select **Use local computer counters**.
- The Performance Logs and Alerts service stops and does not restart. If a network connection is lost during remote monitoring, or if there is a problem with a counter DLL, this could cause the Performance Logs and Alerts service to shut down. The service is configured to restart only once after the first failure. Thereafter, you need to start the service manually. To avoid recurrence of this problem, modify the startup properties of the services in **Services** under **Administrative Tools**. Also make sure to investigate the cause of the shutdown by reviewing the event log and disabling problem DLLs. To do this, use Exctrlst, a utility included on the *Windows 2000 Resource Kit* companion CD.
- Workspace (.pmw) files that were created with Windows NT 4.0 Performance Monitor are not fully compatible with Windows 2000 System Monitor. Only one of the views saved in the workspace is available in System Monitor. System Monitor can read log files created with earlier versions of Performance Monitor when you use the following syntax at the Windows 2000 command prompt:

  **perfmon.exe** *log_file_name*

This command does not invoke Windows NT 4.0 Performance Monitor, but instead a shell program that starts System Monitor. Windows NT 4.0 Performance Monitor is available as Perfmon4.exe on the *Windows 2000 Resource Kit* companion CD.

## Controlling Performance Monitoring Overhead

When you select a counter in any view, the performance tools collect data for all counters of that object, but display only the one you select. This causes only minimal overhead, because most of the tools' overhead results from the display. You can control monitoring overhead in the following ways:

- Use logs instead of displaying a graph. The user interface is more costly in terms of performance.
- Limit the use of costly counters; this increases monitoring overhead. For information about costly counters, see the Performance Counter Reference on the *Microsoft® Windows® 2000 Resource Kit* companion CD.
- Lengthen collection intervals if possible. In general, 600-second (10-minute) intervals are sufficient for ordinary monitoring.
- Collect data during peak activity rather than over an extended interval.
- Reduce the number of objects monitored unless these are critical to your analysis.
- Put the log file on a disk that you are not monitoring.
- Check the log file size when logging multiple servers to a single computer to see how much space the data is taking up.
- Limit to brief periods the trace logs that are monitoring page faults or file I/O. Prolonged trace logging strains system performance.
- Avoid configuring System Monitor reports to display nondefault data. If you choose nondefault data (the defaults are Average value for logs; Last value for graphs) in the Report view, the statistic is calculated at each sample interval. This incurs some additional performance overhead.

# Specific Monitoring Scenarios

This section describes various scenarios such as monitoring remote computers and monitoring servers and services.

## Monitoring Remote Computers

In general, monitoring remote computers differs little from monitoring local computers. This section discusses some facts to consider when evaluating whether to monitor remotely or locally.

## Methods of Monitoring

When monitoring activity on remote computers, you have some options with regard to how to collect data. For example, you could run a counter log on the administrator's computer, drawing data continuously from each remote computer. In another case, you could have each computer that is running the service collect data and, at regular intervals, run a batch program to transfer the data to the administrator's computer for analysis and archiving. Figure 5.12 illustrates these options.

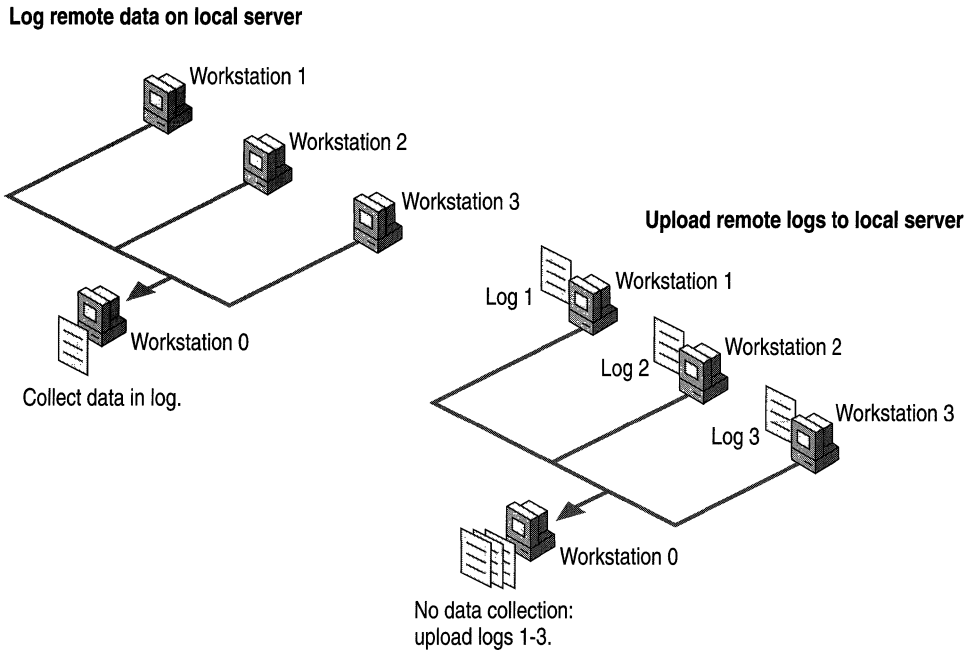



**Figure 5.12 Comparison of Performance Data Logging Options**

Choose a monitoring method based on your needs from the ones described in the following list:

- Centralized data collection (that is, collection on a local computer from remote computers that you are monitoring) is simple to implement because only one logging service is running. You can collect data from multiple systems into a single log file. However, it causes additional network traffic and might be constrained by available memory on the administrator's computer. Frequent updating also adds to network activity. Centralized monitoring is useful for a small number of servers (25 or fewer). For centralized monitoring, use the **Add Counters** dialog box to select a remote computer while running System Monitor on your local computer.

- Distributed data collection (that is, data collection that occurs on the remote computers you are monitoring) does not incur the memory and network traffic problems of local collection. However, it does result in delayed availability of the data, requiring that the collected data be transferred to the administrator's computer for review. This kind of monitoring might be useful if you suspect the server is part of the problem. It is also useful if you suspect that the network is the cause of performance problems and you are concerned that data packets you want to monitor are being lost, because it isolates the computers from the network during data collection. In general, local monitoring creates more disk traffic on each monitored computer. For distributed monitoring, use **Performance Logs and Alerts** under **Computer Management** to select the computer you want to monitor.

## Security Issues

If you are collecting data using the registry, monitoring a remote computer requires the use of the Remote Registry Service. If the service stops due to failure, the system restarts it automatically only once. Therefore, if the service stops more than once, you must restart the service manually on the second and any subsequent failures. To change this default behavior, modify the properties for Remote Registry Service. You can access service properties using **Services** under **Services and Applications** in **Computer Management** or under **Administrative Tools**. Also check the application and system logs in Event Viewer for events that might explain why the service stopped.

In addition, remote data collection requires access to certain registry subkeys and system files. Users need a minimum of Read access to the Winreg subkey in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurePipeServers to provide remote access to the registry for the purpose of collecting data on remote systems. By default, members of the Administrators group have Full Control access and members of the Backup Operators group have Read access. Users also need Read access to the registry subkey that stores counter names and descriptions used by System Monitor. This subkey is HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib*LanguageID*, where *LanguageID* is the numeric code for the spoken language for the operating system installation. (For the English language, the subkey is Perflib\009.) By default, members of the Administrators and Creator Owners groups, and the System account, have Full Control access and members of the Interactive group have Read access.

Users might also require read access to the files that supply counter names and descriptions to the registry, Perfc*.dat and Perfh*.dat. (The asterisk is a wildcard character representing the specific language code; for English, these are Perfc009.dat and Perfh009.dat.) If these files reside on an NTFS volume, then, in order to have access to them, the access control lists (ACLs) on these files must specify that the user has such access. By default, members of the Administrators and Interactive groups have sufficient access.

The remote computer allows access only to user accounts that have permission to access it. In order to monitor remote computers, the Performance Logs and Alerts service must be started in an account that has permission to access the remote computers you are attempting to monitor. By default, the service is started under the local computer's system account, which generally has permission to access only services and resources on the local computer. To start this under a different account, start Computer Management, click the plus sign (+) beside **Services and Applications**, and click **Services**. Click **Performance Logs and Alerts**, and update the properties under the **Log On** tab. To monitor using counter logs or alerts, you must also have permission to read the HKEY_CURRENT_MACHINE\SYSTEM \CurrentControlSet\Services\SysmonLog\LogQueries registry subkey.) In general, administrators have this access by default. In each case, attempting to use the tools without appropriate permissions will generate an error message.

If you are collecting data remotely by means of WMI, the user must be a member of the Administrators group.

# Monitoring Servers and Services

Computers that provide shared resources to users of a network are called servers. This section describes the relationship between a the workload of a server and its resource utilization. This section also covers establishing a baseline for a server, identifying emerging problems, and upgrading.

## Characterizing Server Workload

Some servers are characterized by the workload they support, as indicated in Table 5.9. Servers supporting these workloads can have specific resource requirements. These requirements and how to monitor them with performance monitoring tools are described later in this chapter.

**Table 5.9 Objects to Monitor Based on Server Workload**

| Server workload | Heaviest resource usage | Objects to monitor |
|---|---|---|
| Application servers | Memory and processor | Cache, Memory, Processor, and System |
| Servers used for backups | Processor and network | System, Server, Processor, and Network Segment |
| Database servers | Disks and processor | PhysicalDisk, LogicalDisk, Processor, and System<br>If using Microsoft® SQL Server™, see the product documentation for information about installed performance objects. |

*(continued)*

**Table 5.9 Objects to Monitor Based on Server Workload *(continued)***

| Server workload | Heaviest resource usage | Objects to monitor |
|---|---|---|
| Domain controllers | Memory, processor, network, and disk | Memory, Processor, System, Network Segment, Network Interface, protocol counters (TCP, UDP, ICMP, IP, NBT Connection, NetBEUI, NetBEUI Resource, NWLink IPX, NWLink NetBIOS, NWLink SPX), PhysicalDisk, and LogicalDisk<br>The NWLink object counters report zero values for frame-related data.<br>For Active Directory, also monitor the NTDS and Site Server LDAP Service objects; for Windows 2000 servers, monitor the Browser object. As applicable, monitor DNS or WINS objects. |
| File and print servers | Memory, disk, and network components | Memory, Network Segment, PhysicalDisk, and LogicalDisk<br>For print servers, use the Print Queue object for monitoring queue activity. |
| Mail/messaging servers | Processor, disk, and memory | Memory, Cache, Processor, System, PhysicalDisk, and LogicalDisk<br>If using Microsoft Exchange, see the product documentation for information about installed performance objects. |
| Web servers | Disk, cache, and network components | Cache, Network Segment, PhysicalDisk, and LogicalDisk |

**Tip** As a general recommendation, servers with major roles should be dedicated to a single purpose rather than shared among multiple purposes. For example, do not have domain servers or database servers do double duty as application or file and print servers because these secondary activities can impede the ability of the server to handle its primary workload.

As a service runs, it makes specific demands on system resources. In addition to the monitoring resource requirements defined by the computer's workload, you can obtain information about the services using performance data made available through the services. Many services running under Windows 2000 Server supply performance counters that can be used to measure service activity and resource utilization. Monitoring these counters provides data you can use to determine how well your server is functioning.

In addition to the standard counters described in "Overview of Performance Monitoring," you can use the System object counters to obtain information of interest on server computers. For example, you might want to examine the values for System\System Up Time. If this value is consistently low, it might indicate frequent failures, meaning that the server remains up only for a short time.

## Establishing a Baseline on Server Computers

Monitor the performance of your server at the busiest times of day as well as when it is idle or when activity levels are low. Consider the number of concurrent users as well as the number of inactive connections during the times you are monitoring because the server expends resources tracking inactive as well as active connections. After collecting and reviewing this data, you can set a baseline for the performance of your server. This baseline can be used as a reference point for determining when performance problems exist and for projecting future resource requirements. When data consistently falls outside of the baseline ranges, consider it a signal that the performance of your server is changing—either because the demand is increasing or a bottleneck is developing. You will then want to monitor activity more closely to determine whether your server needs tuning or upgrading.

When you become aware of changes in performance that result in values outside your baseline, monitor the system counters representing the disk, memory, processor, and network components along with the Processor\% Processor Time counter for the services that you are running to see if there is a correlation between service activity and resource usage. If so, you might need to tune the service.

---

**Note** Unless you are testing a computer's startup performance, wait until the startup process has concluded before monitoring system performance. Otherwise, you might mistake the high utilization values characteristic of process startup for the process's typical load.

---

In general, you will want to monitor a server from a different computer so as to minimize the impact of the testing process on the computer being tested. For more information about the advantages and disadvantages of remote versus local monitoring, see "Monitoring Remote Computers" earlier in this chapter.

## Upgrading Servers and Capacity Planning

The chapters "Analyzing Processor Activity," "Evaluating Memory and Cache Usage," and "Examining and Tuning Disk Performance" in this book provide information about tuning and planning for upgrades for hardware resources such as memory, processors, or disks when counter values indicate that these are being strained by your workload. In addition, the chapter "Measuring Multiprocessor System Activity" addresses the special considerations for obtaining optimal performance from scaled systems and discusses the benefits of server clustering and load balancing.

The capacity of a server depends on many variables, including the number of work units (such as client requests), the amount of time required for each unit, and the corresponding resource utilization. Because of the many variables of an individual server implementation, it is extremely difficult to arrive at a perfect formula for gauging the capacity of your server.

# Monitoring Legacy Applications

For optimal performance, it is recommended that all applications you run under Windows 2000 be 32-bit. However, if you need to continue using 16-bit Windows or MS-DOS-based applications, this section describes how you can monitor their activity.

## Monitoring 16-bit Windows Applications

In Windows 2000, by default, all active 16-bit Windows applications run as separate threads in a single multithreaded process called NT Virtual DOS Machine (NTVDM). The NTVDM process simulates a 16-bit Windows environment complete with all of the DLLs called by 16-bit Windows applications.

This configuration poses two challenges for running 16-bit applications:

- It prevents 16-bit applications from running simultaneously, which might impede their performance.
- It makes monitoring a bit trickier because 16-bit applications do not appear by name in the **Add Counters** dialog box for the performance tools; instead, they appear as undistinguishable NTVDM processes.

As a result, Windows 2000 includes an option to run a 16-bit application in its own separate NTVDM process with its own address space. You can monitor 16-bit Windows applications by identifying them by their thread identifier (ID) while they are running, or by running each application in a separate address space.

In addition to the 16-bit applications, each NTVDM process includes a heartbeat thread that interrupts every 55 milliseconds to simulate a timer interrupt, and the Wowexec.exe thread, which helps to create 16-bit tasks and to handle the delivery of the 16-bit interrupt. This thread supports 16-bit Windows applications in a 32-bit Windows environment. The WOW subsystem provides a virtual DOS machine where all Win16 applications run. You will see the heartbeat and Wowexec threads when monitoring 16-bit applications.

Only one 16-bit Windows application thread in an NTVDM can run at one time and, if an application thread is preempted, the NTVDM always resumes with the same thread. This limits the performance of multiple 16-bit applications running in the same NTVDM process, although this limitation becomes an issue only when the processor is very busy.

Although System Monitor can monitor 16-bit applications in Windows 2000 because they run in the same process, the trick to monitoring more than one 16-bit application is to distinguish among the threads of the NTVDM process.

To monitor one 16-bit application, select the NTVDM process in System Monitor. (Other performance tools used to monitor processes can be used for monitoring the NTVDM process. For more information, see "Analyzing Processor Activity" in this book.) If you have multiple 16-bit processes running in NTVDM, you can distinguish them by their thread identifiers (IDs). You might have to start and stop the 16-bit process to determine which ID is associated with which 16-bit process.

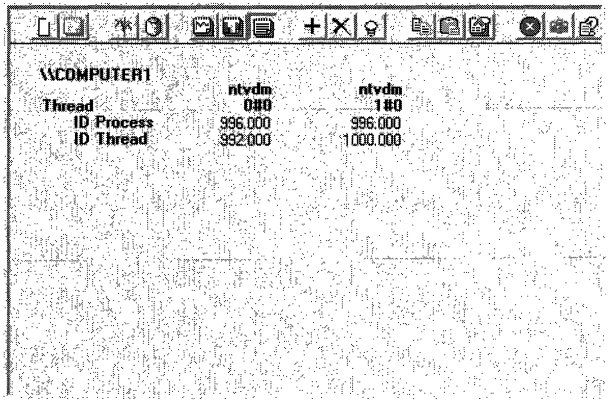



**Figure 5.13 NTVDM Threads in System Monitor**

Figure 5.13 shows a System Monitor report on a single NTVDM process. The ID Process and ID Thread counters are included to help you distinguish among the threads. One of the threads is the heartbeat thread, one is the Wowexec thread, and one is a 16-bit application.

System Monitor identifies threads by the process name and a thread number. The thread numbers are ordinal numbers (beginning with 0) that represent the order in which the threads started. The thread number of a running thread changes when a thread with a lower number stops, and all threads with higher numbers move up in order to close the gap. For example, if thread 1 stopped, thread 2 becomes thread 1. Therefore, thread numbers are not reliable indicators of thread identity.

Instead, with System Monitor you can track processes and threads using the process IDs and thread IDs. The process ID identifies the process in which the thread runs. The thread ID identifies the thread. Unlike the thread number, which can change over the time the thread runs, the system assigns the thread ID when the thread starts and retains it until the thread stops.

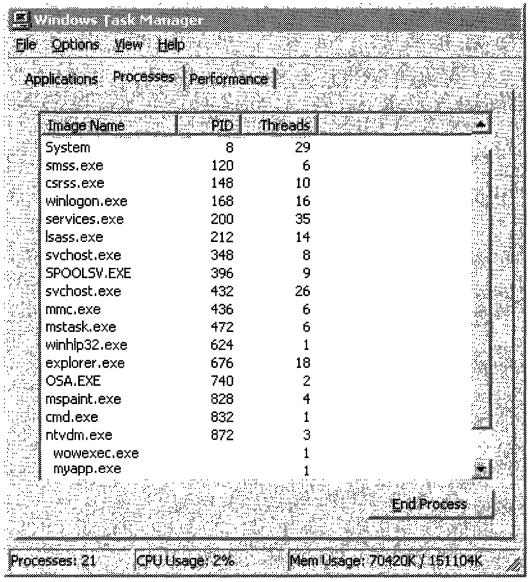



**Figure 5.14 MyApp, Wowexec, and Ntvdm on the Task Manager Process Tab**

As shown in Figure 5.14, Task Manager makes it easy to identify 16-bit applications because it displays the names of the executable files indented below the NTVDM process name. To monitor 16-bit processes in Task Manager, click the **Processes** tab, and on the **Options** menu, click **Show 16-bit Tasks**.

In this example, you can see the Wowexec (Windows on Windows) and the MyApp threads. The heartbeat thread is not an executable and does not appear as a process in Task Manager. However, the **Thread Count** column on the far right shows that all threads are running in the NTVDM process.

# Running 16-bit Windows Applications in a Separate Process

Windows 2000 lets you opt to run a 16-bit Windows application in separate, unshared NTVDM process with its own memory space. This eliminates competition between NTVDM threads in a single process, making the 16-bit application thread fully multitasking and preemptive. It also simplifies monitoring.

▶ **To run a 16-bit application in its own address space**

- At the command prompt, type:

  **start /separate** *processname*

In Task Manager and System Monitor, two instances of the NTVDM process appear. You can use their process identifiers to distinguish between them. Figure 5.15 shows NTVDM threads with the process identifier.

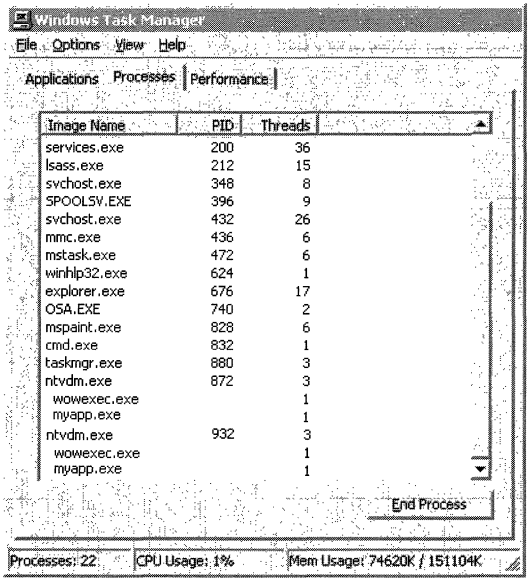



**Figure 5.15 NTVDM Instances in Task Manager with Process Identifiers**

Figure 5.15 shows Task Manager monitoring two copies of a 16-bit application, each in its own NTVDM process.

# Monitoring MS-DOS Applications

In Windows 2000, each MS-DOS application runs in its own NTVDM process, eliminating some of the problems encountered in 16-bit Windows applications. All of the NTVDM processes are called Ntvdm.exe by default, but you can use the following procedure to change the name for easier tracking.

▶ **To create a new process name for an NTVDM process**

1. Copy Ntvdm.exe to a file with a different name.
2. Change the value of the **cmdline** entry in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\WOW to the name of your copy of Ntvdm.exe. The default value is *systemroot*\System32\Ntvdm.exe.
3. When you start an MS-DOS application, it will run in a process with that name. Figure 5.16 shows how your edited process name appears in Regedt32.

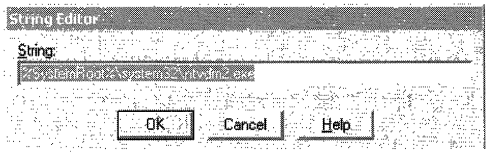



**Figure 5.16 New Process Name in Regedt32**

**Tip** You do not have to restart the computer for the registry change to take effect. Thus, you can change the registry between starting different MS-DOS applications and have each start in a uniquely named process. It is also prudent to set the process name back to Ntvdm.exe when you are finished.

If you are not satisfied with the performance of your MS-DOS-based applications in Windows 2000, try changing the following settings, accessed by right-clicking the file in Windows Explorer:

- Under **Usage** in **Screen** properties for the .pif file, select **Full-Screen** to speed up video display performance. Press ALT+ENTER to get in and out of full-screen mode.
- Disable the Compatible Timer Hardware Emulation feature in the _Default.pif or the application's program information file (PIF). To disable it, clear the check box displayed when you click the **Windows NT** button under **Program** properties for the file. Because this feature causes a decrease in performance, use it only if it is required to allow an application to run under Windows 2000.
- If the application is in a window and seems to pause periodically, try reducing idle sensitivity by moving the **Idle Sensitivity** slider to the left in **Misc** properties for the application's .pif.

- If the MS-DOS-based application can be configured for printing, choose LPT1 or LPT2 rather than parallel port. Most of the applications use Int17 to print when configured for LPT. If you select parallel port mode, these applications print directly to printer ports. Parallel mode is significantly slower in Windows 2000 compared to Windows 3.*x*.

# Integrating the System Monitor Control into Office and Other Applications

You can host the System Monitor ActiveX control in applications of the Microsoft Office 97 suite or later and in HTML pages. The following sections describe how to integrate and use these controls, including the following:

- Inserting the control into an Office document or on a Web page
- Formatting the control within a document

## Placing the Control in an Office Document or on a Web Page

Integration of the Microsoft Office applications with Visual Basic for Applications makes the process of adding the System Monitor control almost identical across these applications. The following procedures describe how to insert the control in applications such as Microsoft Word, Microsoft Excel, and Microsoft PowerPoint®. For purposes of this section, the word "document" is used to refer to PowerPoint slides, Word documents, or Excel spreadsheets.

- Default key settings in Microsoft Word might conflict with the CTRL+H combination used for System Monitor highlighting. You might need to change these to support highlighting when the System Monitor control (Sysmon.ocx in the *systemroot*\System32 folder) is used in Microsoft Word.

In addition, you can add the control to a Web page using an HTML editor that supports insertion of ActiveX controls or using a text editor as described in "To insert the control in an HTML Page."

▶ **To insert the System Monitor control into a Microsoft Office application**

1. Start the application and select **Toolbars** under the **View** menu.
2. Under **Toolbars**, select **Control Toolbox**.

3. With the Control Toolbox displayed, click the **More Controls** icon and select the **System Monitor Control**.

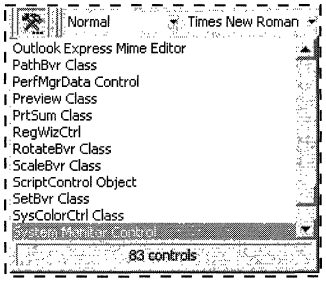



**Figure 5.17 System Monitor Control in the Control Toolbox**

Figure 5.17 shows the System Monitor control selected in the Control Toolbox.

4. Place the control on the page according to the requirements of the application. See the following differences among applications:
   - In a Microsoft Word document, select the insertion point (where the I-beam is flashing) before selecting the control. The control will appear at the position you selected. Design mode is active and you need to exit Design mode before setting control properties or adding counters. Click the **Design Mode** icon to exit.
   - In a Microsoft Excel spreadsheet, select the control and then select the insertion point. When the crosshair appears, note that you can create a placeholder for the control by holding down the left mouse button and dragging the mouse pointer across the columns and rows that you want the control to occupy. Releasing the mouse button causes the control to appear in the space you selected.

     If you do not create a placeholder, you can click any location on the spreadsheet and the control will be inserted in the location with a default size. Design mode is active and you need to exit Design mode before setting control properties or adding counters. Click the **Design Mode** icon to exit.
   - In a Microsoft PowerPoint slide, select the control and then select the insertion point. With the crosshair visible on the screen, note that you can hold down and drag the mouse button to the size you want and the control will be inserted in the placeholder. Otherwise, you can click anywhere on the slide to place the control in that location with a default size and you can modify the size coordinates later (as described in "To format the control in a document").

The Design mode icon is not displayed after you insert the control although Design mode *is* in effect. To exit Design mode, on the **View** menu, click **Slide Show**.

---

**Note** The System Monitor control is not displayed correctly until you exit Design mode.

---

With the control sited in the document, you can add counters or modify control properties, as described in the System Monitor Help file (Sysmon.chm).

If you select **Graph** for the Display Type and the size you define for the control is insufficient to show the graph data, only a portion of the graph will be displayed (such as the vertical minimum and maximum scale values, and the counter-data graph lines).

If you select **Report** for the Display Type and the size is insufficient to show all the report data, the control includes vertical and horizontal scroll bars so that you can scroll to view the data that does not fit in the display.

If you want to format the control in the application, resume Design mode by clicking the Design mode icon in the Control Toolbox and see the procedure named "To format the control in a document" later in this chapter.

▶ **To insert the control in a Microsoft Visual Basic program**

1. On the **File** menu, choose **New** to open a new project (such as a Standard EXE or an ActiveX Document EXE).
2. On the **Project** menu, click **Components** to show the **Components** dialog box.

   You can also view the **Components** dialog box by right-clicking the Toolbox. If the Toolbox is not displayed, in the **View** menu, select the **Toolbox** command.
3. Under **Controls** in the **Components** dialog box, select the **System Monitor Control Library** check box, and then click **OK**.

   An icon for the System Monitor control now appears in the Toolbox.
4. With the form or the document displayed, double-click the **System Monitor Control** icon to add it to the form or document for your program.

   This places the control in the document, occupying the entire size of the form. If you want to select a size for the control, click the **System Monitor Control** icon *once* and then, when the crosshair appears, hold down and drag the mouse to the size you want the control to occupy on the form.

Immediately after inserting the control, the application is in Design mode, so you can format the control within a document. For more information, see the procedure "To format the control in a document" later in this chapter.

▶ **To insert the control in an HTML page**

- Using a text editor, create a page to include the control. To place the control on the page, insert the <OBJECT> tag and specify the class ID as follows:

```
<OBJECT classid="clsid:C4D2D8E0-D1DD-11CE-940F-008029004347">
</OBJECT>
```

This places the control in the document and, when you view the page in a Web browser, you can add counters or modify control properties as described in the procedures that appear later in this document.

To include functions in your page that automate the setting of properties or addition of counters, use scripts written in Visual Basic Scripting Edition (VBScript), or use Microsoft FrontPage®, as described in the following procedure.

▶ **To insert the System Monitor control into a Web page created with Microsoft FrontPage**

1. In a new or existing FrontPage Web page, on the **Insert** menu, select **Other Components**, and then select **ActiveX Control**.
2. In the **ActiveX Control Properties** dialog box, select **Sysmon Graph Control** in the **Pick A Control** list box.
3. To add VBScript, click **Extended**, and complete the **Extended Attributes** dialog box as needed.

   For more information about VBScript, see the VBScript link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

# Formatting the Control in a Document

The default appearance of the control can vary based on the document in which it resides. This is because the control takes the ambient properties of the document in which it is running. For example, when placed in Microsoft Word or FrontPage, the graph background, font, and chart background color might be reset to the default color and font used by those documents. Therefore, after you insert the control in a Microsoft Office application, you might want to change its appearance with respect to the document itself.

Using Design mode in Visual Basic or Office documents, you can easily manipulate the control's attributes such as size, position, and so on as you could with any linked or embedded object. In addition, you have access to the control's properties, methods, and events from the Visual Basic editor for programming the control's behavior within the document. (The control's user interface is inactive when the document is in Design mode.) Note that there are slight differences in how some of the Office applications handle formatting the inserted control.

---

**Note** When changing the properties of the control, it is possible to set colors for **BackColor**, **ForeColor**, or **GraphBackground**, or to set graph line colors that are not visible. Therefore, make sure to check the appearance of your control display after you make changes to its properties.

In addition, you cannot programmatically format the properties and methods of the control using the Visual Basic editor. Use VBScript or Jscript® in your HTML editor for this purpose.

---

▶ **To format the control in a document**

1. Resume Design mode and make the changes you want. To resume Design mode in PowerPoint, exit the Slide Show view.
   - To change the position of the control in the document, drag the control to another position in the container.
   - To change the height or width of the control in the document, drag one of the selection handles of the control in the appropriate direction.

2. To access the control properties or the Visual Basic code editor, right-click the control and select **Properties** or **View Code**, as appropriate. In the Visual Basic code editor window, you can view the Object Browser if needed.

   Notice that some commands on this shortcut menu (such as Cut, Copy, or Paste) are designed for other embedded or linked objects in that document; they are not relevant to the control and might be unavailable (dimmed).

   **Note** When you use Microsoft Office applications, it is possible to modify the control in the document so that the control's properties become inaccessible. This occurs because the control is being converted into an embedded object. For example, if you click the **System Monitor Control Object** command in Microsoft Word (by right-clicking the control) and select the **Convert** command, the **Convert** dialog box appears. If you click **OK** in this box, the control's properties and view-code option become unavailable. You will not be able to change the position or the size of the object in Word.

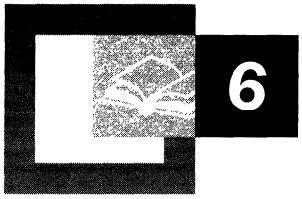
3. Exit Design mode before trying to add counters or to edit the control's default properties.

Notice that the appearance of the control might not update properly until you exit Design mode.

For information about the control's objects, properties, and so on, see the Software Development Kit (SDK) documentation in the MSDN Library at http://windows.microsoft.com/windows2000/reskit/webresources.

CHAPTER 6

# Evaluating Memory and Cache Usage




Use the Performance console and other Microsoft® Windows® 2000 tools to assess available memory and to observe the effects of a memory shortage, a common cause of poor computer performance. Examine the effectiveness of the file system cache—an area of physical memory where recently used data read from or written to the disk is mapped for quick access. In addition, use Windows 2000 tools to investigate memory problems caused by applications that have not been optimized.

### In This Chapter

Overview of Memory Monitoring 291
Determining the Amount of Installed Memory 292
Understanding Memory and the File System Cache 293
Optimizing Your Memory Configuration 296
Establishing a Baseline for Memory 300
Investigating Memory Problems 302

### Related Information in the Resource Kit

- For general information about performance monitoring, see "Overview of Performance Monitoring" in this book.
- For information about developer tools see the *Microsoft Windows 2000 Resource Kit* companion CD, or see the MSDN link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

# Overview of Memory Monitoring

Low memory conditions can slow the operation of applications and services on your computer and impact the performance of other resources in your system. For example, when your computer is low on memory, *paging*—that is, the process of moving virtual memory back and forth between physical memory and the disk—can be prolonged, resulting in more work for your disks. Because it involves reading and writing to disk, this paging activity might have to compete with whatever other disk transactions are being performed, intensifying a *disk bottleneck*. (A disk bottleneck occurs when disk performance decreases to the extent that it affects overall system performance.) In turn, all this work by the disk can mean the processor is used less or is doing unnecessary work, processing numerous interrupts due to repeated *page* CIEBs. (Page faults occur when the system cannot locate requested code or data in the physical memory available to the requesting process.) In the end, applications and services become less responsive.

Figure 6.1 illustrates the sequence in which you conduct the monitoring process. Memory has such an important influence on system performance that monitoring and analyzing memory usage is one of the first steps you take when assessing your system's performance.

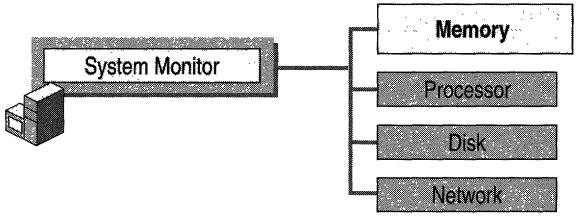



**Figure 6.1 Role of Memory Monitoring in Overall Monitoring Sequence**

The first phase of analyzing memory usage involves understanding your current memory configuration and your workload, using the steps in the following list:

- Determine the amount of physical memory that is currently installed, and compare it against minimum operating system requirements.
- Read "Understanding Memory and the File System Cache" in this chapter for information about the relationship between system memory and the file system cache.
- Establish a baseline for memory usage on your computer by determining performance ranges for low or idle, average, and peak usage periods.
- Optimize the system memory configuration to the system workload, including verifying cache and paging file sizes.

Focus subsequent monitoring on how your system uses memory and on identifying memory shortages or other problems, using the following steps:

- Look at memory characteristics of processes using the Process\Working Set and Process\Private Bytes counters. The Working Set counter reports the amount of committed memory allocated to the process. This might include shared and private bytes currently resident in physical memory. The Private Bytes counter reports memory allocated exclusively to the process. Working set monitoring is important because, when memory is in short supply, the operating system trims the *working sets* of processes and paging occurs. (The working set of a process is the amount of physical memory assigned to that process by the operating system.)
- Monitor the counters listed in "Investigating Disk Paging" later in this chapter to understand the relationship between the amount of paging and the amount of disk activity and their effect on overall performance. Excessive paging can burden the disk.
- Determine the effectiveness of the file system cache. Your system performs better when it can find requested data in the cache, rather than having to read from the disk.
- Learn how to tune the *working sets* of applications, if you have access to source code. Efficient applications maintain a small working set without generating page faults.

# Determining the Amount of Installed Memory

Start your monitoring efforts knowing that you have at least the minimum amount of memory required to run Windows 2000. These requirements are as follows:

- Microsoft® Windows® 2000 Professional: 32 megabytes (MB)
- Microsoft® Windows® 2000 Server: 64 MB

The memory recommendation for Windows 2000 Professional is based on a typical desktop configuration including a business productivity application, such as a word processor or a spreadsheet program, an e-mail application, and a Web browser. The memory recommended for Windows 2000 Server is based on either a dedicated or a multiuse server with a low load, such as a small file-sharing and Web service configuration for a small office. If you are using a database management system such as Microsoft® SQL Server™ or a messaging server such as Microsoft® Exchange, consult the documentation for those products to determine the memory recommended for running them on Windows 2000 Server.

There are a few different ways to determine the amount of memory on your computer. You can find the amount of physical memory installed on your system by clicking the **Performance** tab in Task Manager. Or you can find the amount of available RAM by double-clicking **System** in **Control Panel** and then clicking the **General** tab.

**Note** You can see the memory configuration on local or remote systems using System Information. For more information, see Windows 2000 Server Help.

The operating system distinguishes memory usage by applications and services depending on whether the usage involves the paged or the nonpaged pool. The paged pool contains memory for objects used by applications and services that can be paged to disk; objects in the nonpaged pool cannot be paged to disk. The operating system determines the size of each pool based on the amount of physical memory present. Memory pool usage can be an important factor in evaluating memory usage by your applications. For more information, see "Investigating User-Mode Memory Leaks" and "Investigating Kernel-Mode Memory Leaks" later in this chapter.

The file system cache, which is a subset of physical memory used for fast access to data, and the *disk paging file*, which supports *virtual memory*, influence the amount of memory used by the operating system and applications. (The disk paging file, also called a swap file, is a file on the hard disk that serves as temporary, virtual memory storage for code and data.) Virtual memory is the space on the hard disk that Windows 2000 uses as memory. For purposes of monitoring, the most important types of virtual memory are committed memory that the system sets aside for a process in the paging file and available memory that is not in use by a process. (Another type of memory managed by Windows 2000 is reserved memory, which the system sets aside for a process but which might not be entirely used.) The following sections describe the influence of the cache and the paging file on performance and explain how best to adjust these for optimal memory usage.

# Understanding Memory and the File System Cache

Windows 2000 allocates a portion of the virtual memory in your system to the *file system cache*. The file system cache is a subset of the memory system that retains recently used information for quick access. The size of the cache depends on the amount of physical memory installed and the memory required for applications. The operating system dynamically adjusts the size of the cache as needed, sharing memory optimally between process working sets and the system cache.

On computers running Windows 2000 Server, the value set for the **LargeSystemCache** registry entry in HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager\Memory Management controls the size of the cache. You should set the value of the **LargeSystemCache** entry using the Windows 2000 Server user interface, rather than by editing the registry. Use the **Server Optimization** tab in the **File and Printer Sharing for Microsoft Networks Properties** dialog box to control memory buffer allocation for network connections and the size of the file system cache working set. Figure 6.2 illustrates the user interface for configuring these settings. Notice that these settings are not available on Windows 2000 Professional.

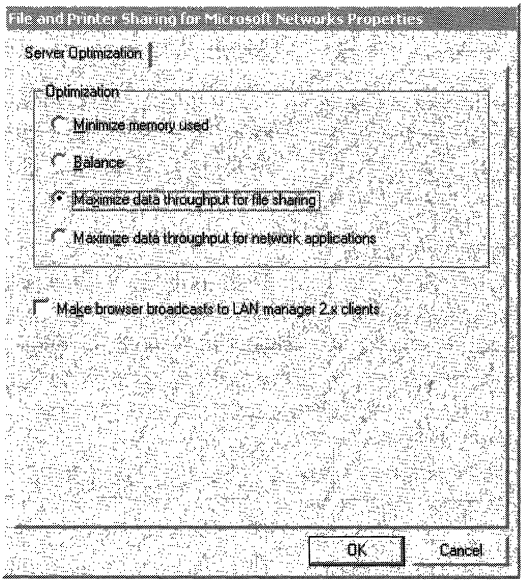



**Figure 6.2 File and Printer Sharing for Microsoft Networks Properties Dialog Box**

▶ **To view or change Server Optimization settings**

1. From the **Start** menu, point to **Settings**, and then click **Control Panel**.
2. Double-click **Network and Dial-up Connections**.
3. Double-click **Local Area Connection**, and then click the **Properties** button.
4. In the **Local Area Connection Properties** dialog box, double-click **File and Printer Sharing for Microsoft Networks**.
5. Under **Optimization**, if you are running Windows 2000 Server, the **Maximize Data Throughput for File Sharing** option is selected by default.
6. Click **OK**.

The values for the **LargeSystemCache** entry are shown in Table 6.1, along with their corresponding options under **File and Printer Sharing for Microsoft Networks Properties**.

**Table 6.1 Settings that Manage Cache Size in the Registry and the User Interface**

| Registry value | User interface option | Description | Notes |
|---|---|---|---|
| 0 | **Maximize data throughput for network applications** | Optimizes systems for distributed applications that typically do their own memory caching (such as SQL Server, which sets this option by default). This setting is useful for computers providing application services because it favors the working sets of processes over the working set of the file system cache. | This setting is useful for larger server applications and database servers such as SQL Server that need to maximize process working sets over the file system cache working set. |
| 1 | **Maximize data throughput for file sharing** | Optimizes the system for file and printer sharing resources. This is the default set by Windows 2000 Server Setup.<br>If you have at least 128 MB of RAM, this value results in a very high maximum size for the cache working set.<br>The **Maximize Data Throughput for File Sharing** option is useful for computers that typically run the Server service for file sharing. | On Windows 2000 the file system cache working set can be increased by 464 MB of additional virtual address space (up to 960 MB) if the system has less than 16 GB of RAM, is not configured to start with the /3GB Boot.ini switch, and the **PagedPoolSize** entry is set to a value other than –1 (or 0 on systems with more than 1 GB of RAM), such as 192000000, the system is not running Terminal Services, and the **SystemPages** entry is not set to –1. For more information about how various memory-management registry entries interact, see "Optimizing Your Memory Configuration" later in this chapter. |

To adjust the way memory is allocated on your system, you might want to tune the settings of the preceding registry entries under the Memory Management subkey. The following section describes how to do this.

# Optimizing Your Memory Configuration

The default memory-management settings for your system are optimal in most situations and do not need to be changed. However, the system can provide a much larger working set and additional virtual address space for the system cache if some settings are manually tuned. These settings include the following registry entries in the HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control \Session Manager\Memory Management subkey:

- **LargeSystemCache**. A value of 1 maximizes the size of the system cache's working set and, with PagedPoolSize set to 192000000 and SystemPages set to 0, provides a large virtual address space for the cache, provided that Terminal Services and the /3GB switch are not used. In addition, the maximum working set and large virtual address space for the file system cache are unavailable if Driver Verifier, a tool for testing kernel drivers, is enabled with the special pool. For information about Driver Verifier, see the Driver Development Kits link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.
- **PagedPoolSize**. Changing this to −1 or leaving it at the default of 0 maximizes the size of the virtual address space used for the paged pool. To set aside extended virtual address space for the file system cache, change this value to 192000000, set **LargeSystemCache** to 1, and set **SystemPages** to 0.
- **SystemPages**. Changing this to −1 provides an additional 464 MB of system page table entries for the virtual address space. Notice that making this change prevents expanding the system cache by 464 MB (limiting it to 512 MB) and limits the paged pool size to 192 MB, provided **PagedPoolSize** is not set to −1 to obtain the maximum virtual address space for the paged pool.

Notice that you have the option of changing the settings for **LargeSystemCache** in the user interface (using the **Server Optimization** tab in **File and Printer Sharing for Microsoft Networks** properties). However, you can only change the settings of **SystemPages**, **PagedPoolSize**, and **NonPagedPoolSize** using a registry editor.

---

**Caution** Do not use a registry editor to edit the registry directly unless you have no alternative. The registry editors bypass the standard safeguards provided by administrative tools. These safeguards prevent you from entering conflicting settings or settings that are likely to degrade performance or damage your system. Editing the registry directly can have serious, unexpected consequences that can prevent the system from starting and require that you reinstall Windows 2000. To configure or customize Windows 2000, use the programs in Control Panel or Microsoft Management Console (MMC) whenever possible.

---

If you are using Terminal Services or your system is configured to start with the /3GB switch, these features consume the extra virtual address space that you might otherwise allocate to the system cache or the paged pool and that space is unavailable.

### Do not change the SecondLevelDataCache entry

Some third-party sources have erroneously reported that modifying the **SecondLevelDataCache** registry entry in HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager\Memory Management can enhance system performance. The second level (L2) cache is recognized by the operating system and is fully utilized regardless of the setting of this parameter.

## Adjusting Paging File Size

For virtual-memory support, Windows 2000 creates one paging file called Pagefile.sys on the disk or volume on which the operating system is installed. The default size is equal to 1.5 times the amount of physical memory. A small paging file limits what can be stored and might exhaust your virtual memory for applications. If you are short on RAM, more paging occurs, which generates extra activity for your disks and slows response times for the system.

**Note** Special considerations apply for sizing the paging file on Windows 2000 Advanced Server when you are using 4 gigabyte RAM tuning (4GT). For more information, see Windows 2000 Advanced Server Help.

Because the size and location of paging files can affect your system's performance, you might want to modify these. Also, because maintaining multiple files on multiple drives can improve performance, you might want to add a paging file. Figure 6.3 shows the **Virtual Memory** dialog box, which you use to change your paging file settings. See Windows 2000 Server Help for specific instructions.

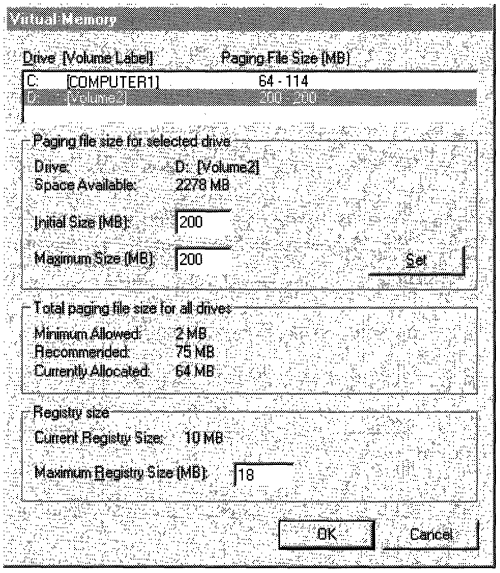



**Figure 6.3 Virtual Memory Dialog Box**

The following guidelines describe how to optimize the paging file.

### Set the Same Initial and Maximum Size

Setting the paging file's initial size and maximum size to the same value increases efficiency because the operating system does not need to expand the file during processing. Setting different values for initial and maximum size can contribute to disk fragmentation.

### Expand the Default Size

Expanding the default size of the paging file can increase performance if applications are consuming virtual memory and the full capacity of the existing file is being used. To determine how large your paging file should be based on your system workload, monitor the Process (_Total)\Page File Bytes counter. This indicates, in bytes, how much of the paging file is being used.

You can also determine the appropriate size of a paging file by multiplying the Paging File\% Usage Peak counter value by the size of Pagefile.sys. The % Usage Peak counter indicates how much of the paging file is being used. You should consider expanding the page file whenever this counter reaches 70 percent of the total size in bytes of all paging files or the Memory\% Committed Bytes In Use counter reaches 85%, whichever occurs first.

A large paging file uses disk storage space, so do not create a large paging file on a disk that is very active (for example, one that services heavy application or network activity) or one that has limited space. Change the file size gradually and test performance until you find the optimal balance between paging file and disk space usage. The operating system requires a minimum of 5 MB of free space on a disk. For more information, see "Examining and Tuning Disk Performance" in this book.

Before increasing the file size, make sure you have adequate disk space, particularly on your servers. For more information, see "Monitoring Network Performance" in this book.

### Move the Paging File

If disk space on your boot volume is limited, you can achieve better performance by moving the paging file to another volume. However, you might want to leave a smaller paging file on the boot volume and maintain a larger file on different volume with more capacity for the sake of recoverability. Depending on how you have configured your system's startup and recovery options, the configuration might require that you maintain a paging file of a certain size on the boot volume. Therefore, make sure to consider your startup and recovery settings when planning to move the paging file. For information about startup and recovery options such as writing debugging information, see Windows 2000 Help.

### Use Multiple Disks

Although Windows 2000 supports a limit of 4,095 MB for each paging file, you can supply large amounts of virtual memory to applications by maintaining multiple paging files. Spreading paging files across multiple disk drives and controllers improves performance on most modern disk systems because multiple disks can process input/output (I/O) requests concurrently in a round-robin fashion.

A mirrored or striped volume is a good candidate for placement of a paging file. Placing the paging file on its own logical partition can prevent file fragmentation. Creating multiple paging files on a single logical volume or partition does not improve performance.

If you find that page writing and disk writing or page reading and disk reading are equivalent on a logical disk, splitting the paging file onto separate volumes is helpful.

### Set a Smaller Default Size for Systems Running Windows 2000 Advanced Server

If you are working in an enterprise environment using the 4-GB RAM Tuning (4GT) option and have a system with more than 4 GB of physical memory, you might want to consider changing the default size of your paging file. In Windows 2000, the default paging file size is equal to 1.5 times the amount of physical memory on the system. On a 4-GB computer, this might result in a paging file too large to be effective. A 256-MB paging file might be a more effective use of disk space. However, a smaller paging size affects the size of total committed memory for applications. For best results, review these settings with the independent software vendor (ISV) for your applications that are being run with the 4GT option. For more information about 4GT, see Windows 2000 Advanced Server Help.

---

**Note** To see how the paging file is used during memory shortages, start the LeakyApp tool on the *Windows 2000 Resource Kit* companion CD, designed to simulate memory leaks for monitoring purposes. While running LeakyApp, monitor Paging File\% Usage and Process(_Total)\Page File Bytes. Log these counters to get an idea of the rate of growth of the paging file.

---

# Establishing a Baseline for Memory

After determining that you have an adequate amount of physical memory and that your configuration is appropriate, examine your physical memory usage under a normal workload to establish a baseline or reference point for physical memory usage. The baseline is generally not a single value but a range within which physical memory usage can fluctuate and still provide acceptable performance. You can use the baseline to identify trends, such as increasing physical memory demands over time, or to recognize problems that arise from a sudden change.

To determine a baseline for your system, use the following counters to create logs of memory usage over an extended period (from several weeks to a month).

- \Memory\Pages/sec
- \Memory\Available Bytes
- \Paging File(_Total)\% Usage

As you monitor the values of these counters, you might see occasional spikes. Typically, you can exclude these from your baseline because it is the consistent, repetitive values with which you are most concerned; the range of values that seem to appear consistently constitutes your baseline. When values fall outside of these ranges for extended periods, follow the instructions provided in this chapter to investigate the variations.

# Virtual Memory Usage

Even if your system exceeds the minimum physical memory requirements for the operating system, you might face situations in which you do not have enough physical memory. For example, if you run several memory-intensive applications or if several users share your computer, the available physical memory of your system could be consumed, affecting your system's performance.

To see how much virtual memory your Windows 2000 Professional–based computer uses, start all applications and use Task Manager to see the Peak Commit Charge value. This value appears in the **Commit Charge** box on the **Performance** tab. Commit charge is the number of pages reserved for virtual memory that are backed by the paging file.

Peak committed memory is the highest amount of virtual memory (in bytes) that has been committed over this sample. To be committed, these bytes must either have a corresponding amount of storage available on disk or in main memory. Compare this value against the size of the paging file to determine if the paging file is sized appropriately.

On a computer running Windows 2000 Server, in addition to total committed memory, you need to consider the number of users sharing a system and the number of files they open to determine memory requirements for your workload.

Under **Computer Management**, use **Shared Folders** under **System Tools** to view this information.

For information about additional recommendations pertaining to memory on server computers, see "Monitoring Network Performance" in this book.

# Default Services Memory Consumption

In general, Windows 2000 has been optimized so that only the most commonly used services run by default, and you should not have to turn off any services. However, you can reduce the memory requirements of your system by turning off some of the default services provided by the operating system.

▶ **To stop a service**

1. From the **Start** menu, point to **Programs**, point to **Administrative Tools,** and then click **Services**.
2. Right-click the name of a service, and then click **Stop**.

This procedure stops the service for the current session. To disable the service permanently, you need to change the value for service start-up in the properties dialog box for the service. To use this dialog box, click Services in the Administrative Tools menu or under the Computer Management console. Right-click the service you want to change, select Properties in the shortcut menu, and change the value to Disabled in the Startup type box.

# Investigating Memory Problems

After you have observed memory usage under normal conditions and established your memory baseline, you might notice that the memory counters sometimes stray from the typical range. The following sections describe how to investigate conditions that cause memory values to deviate from the baseline.

The following activities help you to learn about and analyze memory usage and memory bottlenecks using System Monitor counters and other tools.

- Investigating memory shortages
- Investigating disk paging
- Investigating user-mode memory leaks
- Investigating kernel-mode memory leaks
- Monitoring the cache
- Resolving memory and cache bottlenecks

Table 6.2 summarizes the most important counters to monitor for analyzing memory usage.

**Table 6.2 Counters for Analyzing Memory Usage**

| To monitor for | Use this *Object\Counter* |
|---|---|
| Memory shortages | Memory\Available Bytes or Available KBytes (to see the amount in kilobytes) or Available MBytes (to see the amount in megabytes)<br>Process (*All_processes*)\Working Set<br>Memory\Pages/sec<br>Memory\Cache Bytes |

*(continued)*

**Table 6.2 Counters for Analyzing Memory Usage *(continued)***

| To monitor for | Use this *Object\Counter* |
|---|---|
| Frequent hard page faults | Memory\Pages/sec<br>Process (*All_processes*) \Working Set<br>Memory\Pages Input/sec<br>Memory\Pages Output/sec |
| Excess paging with a disk bottleneck | Memory\Page Reads/sec<br>Physical Disk\Avg. Disk Bytes/Read |
| Paging file fragmentation | PhysicalDisk\Split IOs\sec<br>PhysicalDisk\% Disk Read Time<br>PhysicalDisk\Current Disk Queue Length<br>Process\Handle Count |
| Memory leaks; memory-intensive applications | Memory\Pool Nonpaged Allocations<br>Memory\Pool Nonpaged Bytes<br>Memory\Pool Paged Bytes<br>Process(*process_name*)\Pool Nonpaged Bytes<br>Process(*process_name*)\Handle Count<br>Process(*process_name*)\Pool Paged Bytes<br>Process(*process_name*)\Virtual Bytes<br>Process(*process_name*)\Private Bytes |
| Cache Manager efficiency | Cache\Copy Read Hits %<br>Cache\Copy Reads/sec<br>Cache\Data Map Hits %<br>Cache\Data Maps/sec<br>Cache\MDL Read Hits %<br>Cache\MDL Reads/sec<br>Cache\Pin Read Hits %<br>Cache\Pin Reads/sec<br>To identify cache bottlenecks, also use Memory\Pages Input/sec with these counters. |

**Important** The LogicalDisk object counters are not available by default. If you want to monitor the values for these counters, you must first activate the counters by typing **diskperf –yv** at the Windows command prompt.

# Investigating Memory Shortages

Your system can develop a memory shortage if multiple processes are demanding much more memory than is available or you are running applications that leak memory. Monitor the following counters to track memory shortages and to begin to identify their causes.

- **Memory\Available Bytes** indicates how much physical memory is remaining after the working sets of running processes and the cache have been served.
- **Process (*All_processes*)\Working Set** indicates the number of pages that are currently assigned to processes. When there is ample memory, then the working set structures can fill with pages that are not currently needed to do work but that were needed in the past and might be needed in the future. Because there is no memory shortage, the working set structures are not trimmed. As a result the working set approximates pages that have been referenced in a longer period of time. However, when memory is in short supply, the working set might be trimmed. As a result, the working set in that case approximates the number of pages referenced in a much shorter period of time.
- **Memory\Pages/sec** indicates the number of requested pages that were not immediately available in RAM and had to be read from the disk or had to be written to the disk to make room in RAM for other pages. If your system experiences a high rate of hard page faults, the value for Memory\Pages/sec can be high.

To maintain a minimum number of available bytes for the operating system and processes, the Virtual Memory Manager continually adjusts the space used in physical memory and on disk. In general, if memory is ample, working set sizes can increase as needed. If the memory supply is barely adequate or very close to the amount required, you might see the operating system trim some working-set sizes when another process needs more memory—at startup, for example. Figure 6.4 illustrates this situation.

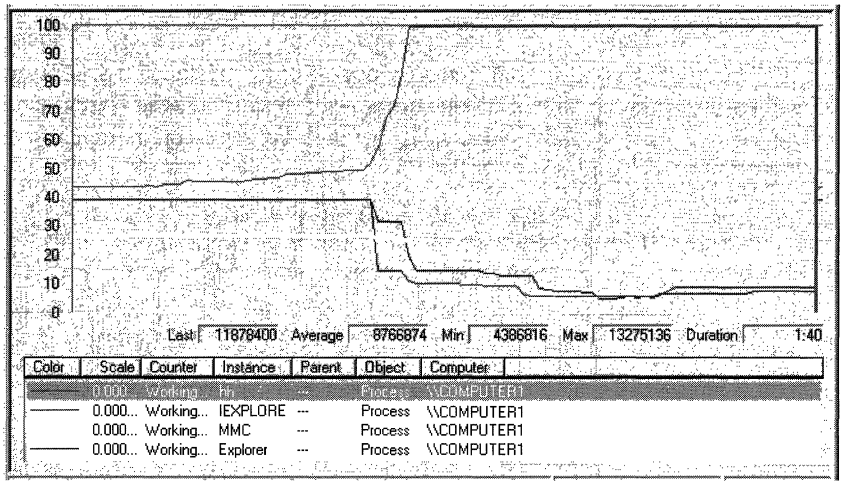



**Figure 6.4 Working Set Values of Processes as One Process Starts Up**

If the value for Memory\Available Bytes is consistently below the system-defined threshold and the value for Memory\Pages/sec spikes continuously, it is likely that your memory configuration is insufficient for your needs. To confirm that a high rate of paging is related to low memory, see "Investigating Disk Paging" later in this chapter.

---

**Note** You might see a low value for Memory\Available Bytes, which is not caused by a memory shortage, in the following situations:

- During large file-copy operations such as a system backup. In this case, you can verify that the copy operation is the cause by monitoring Memory\Cache Bytes as well. You should see Memory\Cache Bytes rise as Memory\Available Bytes falls. Otherwise, you need to investigate the cause as described in this chapter.
- The **Maximize Data Throughput for File Sharing** option is selected in the **File and Printer Sharing for Microsoft Networks Properties** dialog box. This allocates a large system cache, which might not be appropriate for how your computer is used. For more information about how this setting should be configured, see "Optimizing Your Memory Configuration" earlier in this chapter.
- The working sets of processes have become smaller and there is no demand for pages for other purposes. In this case the number of available bytes might be low but there is no need to trim working sets and there are few if any page faults.

---

To identify processes associated with low-memory conditions, examine memory usage by specific processes, and determine if a process is leaking memory as described in "Investigating User-Mode Memory Leaks" and "Investigating Kernel-Mode Memory Leaks" later in this chapter.

If available memory is consistently low (2 MB or less), the computer becomes unresponsive because it is occupied exclusively with disk I/O operations. During paging due to low memory, the processor is idle while waiting for the disk to finish. Therefore, it is important, especially on server computers, to investigate and correct the cause of a low-memory condition. Notice which processes are running as well as the sizes of their working sets as you monitor memory counters. The processes might need to be updated or replaced if they are contributing to memory shortages and you do not want to acquire additional memory.

For information about how to address a memory shortage, see "Resolving Memory and Cache Bottlenecks" later in this chapter.

# Investigating Disk Paging

Use memory counters that report paging activity to identify memory shortages resulting in disk bottlenecks. Start by monitoring the memory counters and working set values as you did when checking for a memory shortage in the preceding section. Confirm that hard page faults are occurring using the Memory\Pages/sec counter.

If hard page faults are occurring, monitor disk counters to assess how the disk is behaving during paging: whether it is busy with other work or with handling page faults. Monitor disk paging using the following steps and associated counters:

1. To confirm hard page faulting use the following Memory and Process counters:
   - Memory\Pages/sec
   - Process (*All_processes*) \Working Set
   - Memory\Pages Input/sec
   - Memory\Pages Output/sec
2. To understand the impact of page faulting on the disk, compare the number of reads and read bytes measured by the following counters:
   - Memory\Page Reads/sec
   - PhysicalDisk\Disk Reads/sec
   - PhysicalDisk\Avg. Disk Read Bytes/sec

## Confirming Hard Page Faults

To confirm hard page faults, examine the hard page fault rate using Memory\Pages/sec and the working sets of active processes using Process(*process_name*)\Working Set, as described in "Investigating Memory Shortages." Memory\Pages/sec is the sum of Pages Input/sec and Pages Output/sec and reports the number of requested pages that were not immediately available in RAM and had to be read from the disk (resulting in hard page faults) or that had to be written to the disk to make room in RAM for other pages. Monitoring processes' working sets enables you to correlate a given application's memory usage with page faulting. When memory is in short supply, working sets are continuously trimmed and page faults are frequent.

Acceptable rates for Memory\Pages/sec range from 40 per second on older laptop computers to 150 per second for the newest disk systems. Use a shorter monitoring period on client computers than on server computers (updating once per second is appropriate) because paging activity can occur in bursts on client computers. Paging activity tends to reach a steady state on server computers; therefore, longer-term monitoring is appropriate.

---

**Note** Page Fault Monitor (Pfmon.exe), a tool on the *Windows 2000 Resource Kit* companion CD, lists hard and soft page faults generated by each function call in a running process. You can display the data, write it to a log file, or both. For more information, see Windows 2000 Resource Kit Tools Help (W2rktools.chm) on the *Windows 2000 Resource Kit* companion CD.

You can also monitor page faults and memory management data using Trace Logs in the Performance snap-in. For more information, see "Overview of Performance Monitoring" in this book.

---

When values exceed the acceptable range for your type of disk, investigate disk activity to determine if the faulting is causing a disk bottleneck. Memory\Pages Input/sec reflects the rate at which pages were read from the disk and thus gives you data on hard page faults. Depending on the capabilities of your disk, high values can indicate a lack of memory sufficient to hurt system performance. See Figure 6.5 for an example of page faulting.

---

**Note** You might see high levels of paging with Memory\Pages/sec when pages are read to and from noncached, memory-mapped files. When these files are used in this way, Memory\Pages/sec or Memory\Available Bytes is high but Memory\Paging File\%Usage and Memory\Cache Faults/sec are normal to low.

---

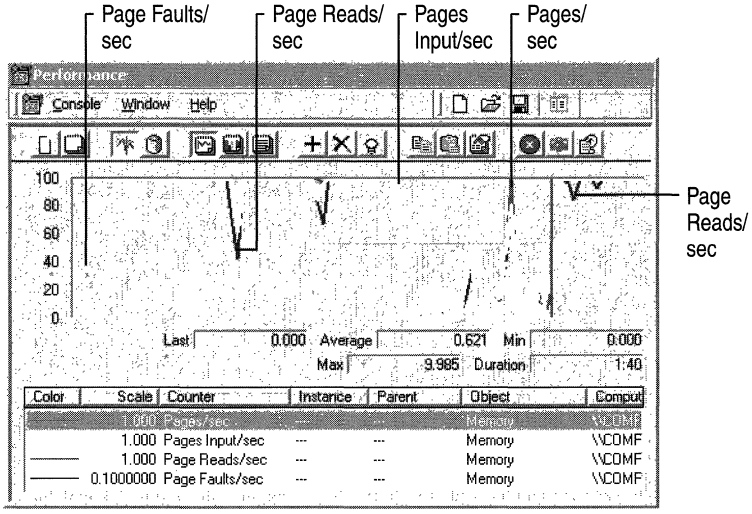



**Figure 6.5 Paging Activity on a System with Low Memory**

The other component of Memory\Pages/sec, Memory\Pages Output/sec, indicates the rate at which pages were written to the disk. Although this activity does not generate hard page faults, disk-write activity can indirectly reflect a memory shortage and indicates additional disk activity. This is because, as the Virtual Memory Manager needs to trim pages from a working set, it might find that some pages contain changed data. In this case the changed data must be written to disk to free the pages. When memory is in ample supply, Memory\Pages Output/sec is likely to be low because there is less need to free changed pages and write that data to disk. For more information about investigating applications that generate disk-write activity, see "Examining and Tuning Disk Performance" in this book.

## Assessing the Effect of Page Faults on the Disk

To understand the impact of page faulting on the disk, examine the number of disk operations that occur as a result of paging. If paging activity dominates your disk's workload, a memory shortage is causing a disk bottleneck. Start by looking at Memory\Page Reads/sec. This counter indicates the number of read operations by the disk that were required to retrieve faulted pages. Compare the number of reads performed with the number of pages faulted to determine how many pages are retrieved per read. A high ratio of reads to faults means a large number of pages are not found in physical memory and are being demanded from the disk, creating a disk bottleneck.

Next, determine what proportion of your disk's overall work is occupied with reading pages from memory. To do this, compare page reads to disk reads. If there is a correlation between the values of Memory\Page Reads/sec and PhysicalDisk\Disk Reads/sec, it is likely that paging activity makes up the majority of your disk activity and could be causing a disk bottleneck.

To see the relationship between paging and disk read operations from a different perspective, monitor the value of PhysicalDisk\Avg Disk Read Bytes/sec or PhysicalDisk\Avg Disk Read Bytes/sec while you are monitoring Page Reads/sec. The Avg Disk Read Bytes/sec counter indicates the rate at which the disk is transferring data during reads. Because this is a measurement of bytes rather than of pages or of the number of reads, you need to convert to identical units. Use the following formula for this purpose: value of PhysicalDisk\Disk Read Bytes/sec ÷ 4096 (number of bytes in a page).

If the result is approximately equal to the value of Page Reads/sec, paging activity is the bulk of your disk read activity, and the memory shortage represented by heavy paging activity could in turn be causing a disk bottleneck. To see if this activity is reaching a rate high enough to cause poor disk performance, see your disk manufacturer's documentation for the number of I/O transactions per second you can expect from your disk. Disks currently available can sustain a transfer rate of 70 I/O operations per second. For more information about locating disk bottlenecks, see "Examining and Tuning Disk Performance" in this book.

Figure 6.6 illustrates disk activity associated with paging activity when memory is low.

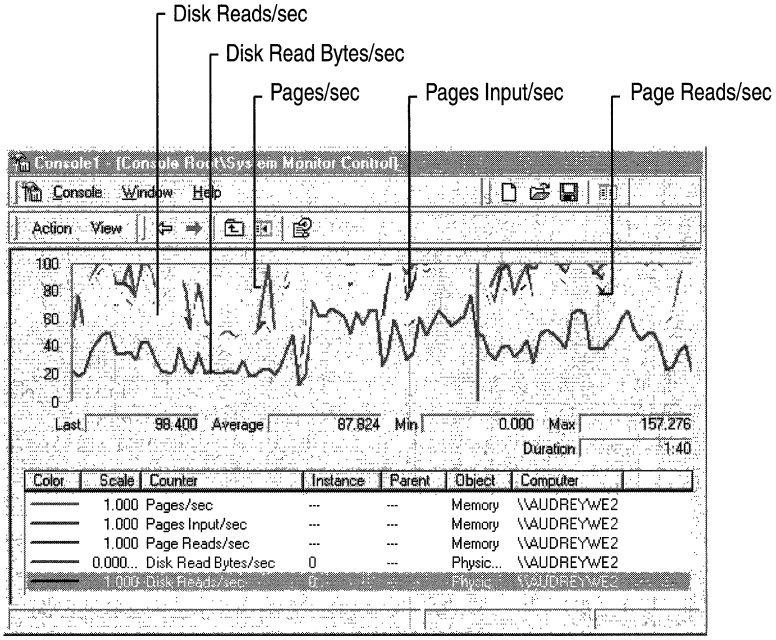



**Figure 6.6 Disk and Paging Activity When Memory Is Low**

# Investigating User-Mode Memory Leaks

A memory leak occurs when applications allocate memory for use but do not free allocated memory when finished. As a result, available memory is used up over time, often causing the system to stop functioning properly. Therefore, it is important to investigate the causes of all memory leaks, particularly on server computers. This section describes how to identify memory leaks, including ones that affect the critical nonpaged memory pool, and where to find tools and information related to working with memory leaks.

---

**Note** It is sometimes possible to mistake an increase in system load for a memory leak. To distinguish between these conditions, observe the Memory and Process counters over a number of days. If you see the system first reach a steady state, then attain a level of increased load (usually achieved during some peak portion of the day), and then fall again, it is likely you are seeing variations in load rather than a leak. On network computers, look at user sessions and throughput rates such as transferred bytes per second to eliminate workload as a factor.

---

## Identifying a Memory Leak

The symptoms of a memory leak include the following:

- A gradually worsening response time.
- The appearance of an error message, shown in Figure 6.7, indicating that the system is low on virtual memory. (Another message box might precede this, indicating that virtual memory has been exceeded and that the system has increased the paging file size automatically.)

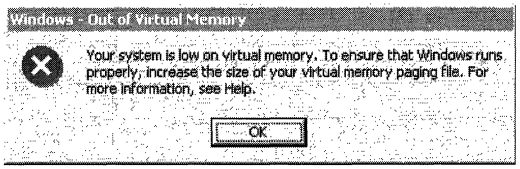



**Figure 6.7 Out of Virtual Memory Error Message**

- The appearance of error messages indicating that system services have stopped.

If you suspect that a particular application or service is causing a memory leak, investigate the memory use of your applications using the following counters:

- **Memory\Available Bytes** reports available bytes; its value tends to fall during a memory leak.
- **Memory\Committed Bytes** reports the private bytes committed to processes; its value tends to rise during a memory leak.
- **Process(*process_name*)\Private Bytes** reports bytes allocated exclusively for a specific process; its value tends to rise for a leaking process.
- **Process(*process_name*)\Working Set** reports the shared and private bytes allocated to a process; its value tends to rise for a leaking process.
- **Process(*process_name*)\Page Faults/sec** reports the total number of faults (hard and soft faults) caused by a process; its value tends to rise for a leaking process.
- **Process(*process_name*)\Page File Bytes** reports the size of the paging file; its value tends to rise during a memory leak.
- **Process(*process_name*)\Handle Count** reports the number of handles that an application opened for objects it creates. Handles are used by programs to identify resources they must access. The value of this counter tends to rise during a memory leak; however, you cannot rule out a leak simply because this counter's value is stable.

Monitor these counters over a period ranging from two hours to a few days. Logging is recommended, both because of the overhead of monitoring multiple instances of the Process counters and because leaks tend to manifest themselves slowly.

In addition, to isolate the problem and avoid unnecessary overhead, monitor from a remote computer, if possible. Network activity or interaction with other computers can interfere with the results.

## Memory Leaks and the Nonpaged Pool

Although any leak is serious, memory leaks are of particular concern when they involve the nonpaged pool. Many system services allocate memory from the nonpaged pool because they need to reference it when processing an interrupt and cannot take a page fault at that time. To identify whether or not a leak affects the nonpaged pool, include the following counters in your monitoring:

- Memory\Pool Nonpaged Bytes
- Memory\Pool Nonpaged Allocs
- Process(*process_name*)\Pool Nonpaged Bytes

---

**Note** Because the internal counters used by Task Manager, Process Monitor, and System Monitor to measure the size of the nonpaged pool for each process are not precise, it is recommended that you monitor changes in the overall pool size over time (a few days, for example), rather than rely on the absolute, instantaneous values reported for each process. The counter values are estimates that count duplicate object handles as well as space for the object. Also, because the process pool size counts are rounded to page size, pool space is overestimated when a process uses only part of a page. In contrast, total pool size counts are precise. Therefore, the sum of pool sizes for each process might not equal the value for the whole system.

---

The counters on the Memory object monitor the total size of the nonpaged pool and the number of allocations of pool space for the whole system. The counter on the Process object monitors nonpaged pool space allocated to each process.

To use System Monitor to monitor the nonpaged pool for leaks, follow these steps:

- Record the size of the nonpaged pool when the system starts. Then log the Memory and Process objects for several days; a 10-minute update interval is sufficient.

- Review the log for changes in size of the nonpaged pool. You should be able to associate any increases in the size of the pool, as indicated by Memory\Pool Nonpaged Bytes, with the start of a process, as indicated by Process\% Processor Time. Also look at individual Process object counters such as Process\Handle Count, Process\Private Bytes, Process\Nonpaged Pool Bytes, Process\Paged Pool Bytes, and Process\Threads. During a memory leak, you might also see rising values for these counters.

  When processes are stopped, you should see a decrease in pool size. Any growth in the nonpaged pool is considered abnormal, and you need to distinguish which process is causing the change in pool size.

  You might also want to monitor the number of active threads before and after running the process (use the **Performance** tab in Task Manager or the Objects\Threads or Process(_Total)\Thread Count counters). A process that is leaking memory might be creating a large number of threads; these appear when the process starts and disappear when the process stops.
- Watch the value of Memory\Pool Nonpaged Bytes for an increase of ten percent or more from its value at system startup to see if a serious leak is developing.

The following additional tools provide information about the paged and nonpaged memory pools as listed in Table 6.3. These tools collect their data from the same sources.

**Table 6.3 Tools That Provide Information About Memory Pools**

| Tool Name | Description | Location |
|---|---|---|
| Memsnap (memsnap.exe)[1] | Records system memory usage to a log file. | Windows 2000 Support Tools |
| Process Monitor (pmon.exe)[1] | Provides total and per process values for nonpaged and paged pool memory. Also monitors the committed memory values shown in the Pmon display for increases; the process with the leak should have an increasing value reported under Commit Charge. | Windows 2000 Support Tools |

[1] These tools are useful because they show allocations on a per-process basis.

For information about installing and using the Windows 2000 Support Tools and Support Tools Help, see the file Sreadme.doc in the \Support\Tools folder of the Windows 2000 operating system CD.

For a quick demonstration of a memory leak, start LeakyApp, a test tool on the *Windows 2000 Resource Kit* companion CD, and observe the values of the monitored counters. Notice the steady increase in the following counters: Memory\Pages/sec, Process(LeakyApp)\Working Set, and Process(LeakyApp)\Private Bytes.

Figure 6.8 illustrates counter activity during a memory leak generated by the LeakyApp tool.

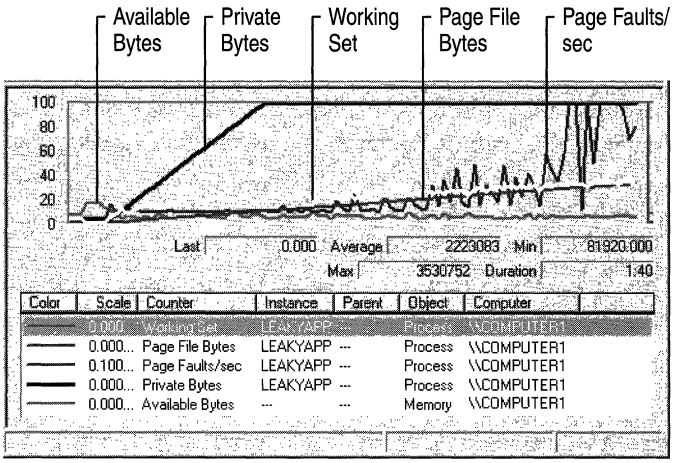



**Figure 6.8 Process Memory Activity During a Memory Leak**

Although the memory leak illustrated in Figure 6.9 has a systemic effect, the problem can be tracked to a single cause—the leaking application. If you have an application that exhibits similar behavior, it is recommended that you either modify it (if you have access to the source code) or replace it with another program.

Developer tools for analyzing and tuning memory usage by applications are available on the *Windows 2000 Resource Kit* companion CD. For more information about developer tools, see the MSDN link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

The following tools optimize memory-intensive applications.

- **Application Monitor (ApiMon)** on the *Windows 2000 Resource Kit* companion CD monitors page faults caused by an application and reports them by Microsoft® Win32® function calls.
- **The Working Set Tuner (WST)** on the SDK analyzes the patterns of function calls in your application code and generates an improved function ordering to reduce physical memory usage. One objective of tuning your working set is to arrive at the smallest possible working set without causing page faults.
- **Virtual Address Dump (Vadump)** on the SDK creates a list that contains information about the memory usage of a specified process, including address size, total committed memory for the image, the executable, each dynamic-link library (DLL), and heap usage.

  The nonpaged pool size and the paged pool size are set by default during Windows 2000 Setup based on your memory configuration. The maximum nonpaged pool size is 256 MB. The maximum paged pool size is approximately 470 MB. The actual size varies depending on your configuration. More physical memory results in less paged pool because the virtual address space must instead be used to contain more critical memory-management structures. The pool sizes are defined in the registry in the HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager\Memory Management subkey.

  To extend the file system cache working set from 512 MB to 960 MB, set the value of the **PagedPoolSize** registry entry to 192000000, set **SystemPages** to 0, and make sure that the system is optimized for file sharing with **LargeSystemCache** set to 1. To make the maximum virtual address space available to the paged pool, set the **PagedPoolSize** registry entry to – 1, provided your system is not using the /3GB Boot.ini switch.

  Typically you do not need to set the **NonPagedPoolSize** entry because on systems with more than 1.2 GB of memory, the system automatically defaults to the maximum nonpaged pool size. If you need to set the **NonPagedPoolSize** value, set it to the value you want (in bytes); do not set it to–1.

# Investigating Kernel-Mode Memory Leaks

Kernel-mode processes such as device drivers can also leak memory when bytes are allocated but not freed. Again, you typically track these over a period of several hours or days, but instead of relying on System Monitor counters, use Pool Monitor (Poolmon.exe). For information about Pool Monitor, see Windows 2000 Support Tools Help. For information about installing and using the Windows 2000 Support Tools and Support Tools Help, see the file Sreadme.doc in the \Support\Tools folder of the Windows 2000 operating system CD.

Pool Monitor (Poolmon.exe) shows the amounts of nonpaged and paged memory that were allocated and freed, calculates the difference between these, and associates the data with a function tag to help you identify the process involved. By default, Windows 2000 is configured not to collect pool information because of the overhead. To use Poolmon, you must enable the pool tag signal. Use Gflags.exe to make the change. In Gflags, select the **Enable Pool Tag** check box. For information about using Poolmon and Gflags, see Support Tools Help. For information about installing and using the Windows 2000 Support Tools and Support Tools Help, see the file Sreadme.doc in the \Support\Tools folder of the Windows 2000 operating system CD.

The pool tag is a mechanism for identifying the driver or other part of the kernel allocated to a particular portion of memory. These tags can be examined to reveal memory leaks and pool corruption, and the offending code component can be determined by finding which code component is allocated to which tag. Look for a tag with rapidly increasing byte counts that does not free as many bytes as it allocates, and verify that this tag corresponds to a function for which the increasing memory allocation might be appropriate. If it does not appear appropriate, it might be necessary to debug and tune the application to eliminate the leak.

# Monitoring the Cache

The Windows 2000 file system cache is an area of memory into which the I/O system maps recently used data from disk. When processes need to read from or write to the files mapped in the cache, the I/O Manager copies the data from or to the cache, without buffering or calling the file system, as if it were an array in memory. Because memory access is quicker than a file operation, the cache provides an important performance boost to the processes.

Cache object counters provide information about data hits and misses and about file I/O operations that reflect how efficiently their applications access data in the file system cache. However, because the cache counters are based on the views mapped by the Cache Manager and not on data from the Virtual Memory Manager, the cache counter values do not provide definitive information about bottlenecks. In some cases, cache counters can report low rates of hits or high rates of misses, suggesting that the system is accessing the disk when, in fact, the requested data has been retrieved from memory. This can occur if virtual addresses mapped by the Cache Manager no longer exist, for example, because the file has closed. To obtain more accurate data on I/O bottlenecks, use the Memory\Pages Input/sec counter.

The cache itself can never really have a bottleneck because it is just a part of physical memory. However, when there is not enough memory to create an effective cache, the system must retrieve more data from disk, resulting in I/O operations that can impact overall performance. Such a shortage of cache space can be considered a cache bottleneck.

Cache bottlenecks are of greatest concern to administrators of Windows 2000 Server–based computers and to users of Windows 2000 Professional–based computers running CAD/CAM applications or large databases accessing large blocks of multiple files that rely on the cache. Administrators of server computers need to monitor cache counters because these provide indirect indications of the memory supply. Windows 2000 Server defines a large system cache by default to maximize the likelihood that data is found in memory, rather than retrieved from disks. The cache counters can reveal that a small cache (based on a short supply of memory) is hurting system performance.

---

**Note** Many Web servers or large database applications, such as SQL Server, maintain a separate cache and do not use the file system cache. In addition, they might have their own cache-monitoring counters.

---

With memory-mapped file I/O, an entire file can be opened without actually being read into memory. Only the views (smaller portions of the file) are mapped into the process's address space just before I/O operations are performed. Memory-mapped I/O keeps the cache size small. For more information about memory-mapped I/O, see the Software Development Kit (SDK) link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

Developers who want to understand how their programs use the cache for read and write operations might also want to monitor the cache. Data requested by an application is mapped to the cache and then copied from there. Data changed by applications is written from the cache to disk by the lazy writer system thread or by a write-through call from the application. The lazy writer thread in the system process periodically writes changed pages from the modified page list back to disk and flushes them from the cache. When a page is flushed from the cache, changed data is written from the page in the cache to the disk and the page is deleted from the cache. Thus, monitoring the cache is like watching your application I/O. Remember, however, that if an application uses the cache infrequently, cache activity has an insignificant effect on the system, on the disks, and on memory.

## Understanding the Cache Counters

Use the following System Monitor Cache and Memory counters to measure cache performance. For instructions based on specific symptoms, see "Interpreting Changes in Cache Counter Values" later in this chapter.

**Cache\Copy Read Hits %** Monitor this counter for hit rates and miss rates. Any value over 80 percent indicates that the application uses the cache very efficiently. Compare this counter with Cache\Copy Reads/sec to see how many hits you are really getting. Even though the hit percentage is small, if the rate of operations is high, this might indicate better cache effectiveness than a high percentage with a low rate of operations.

---

**Note** For best results, monitor the Cache\Copy Read Hits % counter using the graph view rather than the report view. Hit percentage rates frequently appear as spikes that are hard to detect in a report.

---

**Cache\Copy Reads/sec** Observe this counter to see the rate at which the file system attempts to find application data in the cache without accessing the disk. This is a count of all copy read calls to the cache, including hits and misses. Copy reads are the usual method by which file data found in the cache is copied into an application's memory buffers.

**Cache\Data Flush Pages/sec** Review this counter for the rate at which applications change pages of cached data and the pages are written back to disk. This includes pages that have been written by the system process when many changed pages have accumulated, pages that have been flushed so that the cache can be trimmed, and disk writes that are caused by an application write-through request.

**Cache\Data Flushes/sec** Monitor this counter for the rate at which cache data is being written back to disk. This counter reports application requests to flush data from the cache and is an indirect indicator of the volume and frequency of application data changes.

**Cache\Data Maps/sec** Examine this counter for the rate at which file systems map pages of a file into the cache for reading. This counter reports read-only access to file system directories, the file allocation table (FAT) in the FAT file system, and the Master File Table in the NTFS file system. This counter does not reflect cache use by applications.

**Cache\Fast Reads/sec** Observe this counter for the rate at which applications bypass the file system and access data directly from the cache. A value over 50 percent indicates the application is behaving efficiently. Fast reads reduce processor overhead and are preferable to I/O requests.

**Cache\Lazy Write Flushes/sec** Review this counter for the rate at which an application changes data, causing the cache to write to the disk and flush the data. If this value reflects an upward trend, memory might be becoming low. Lazy write flushes are a subset of data flushes. The lazy writer thread in the system process periodically writes changed pages from the modified page list back to disk and flushes them from the cache. This thread is activated more often when memory needs to be released for other uses. This counter counts the number of write and flush operations, regardless of the amount of data written.

**Cache\Lazy Write Pages/sec** Examine this counter value for the rate at which pages are changed by an application and written to the disk. If the counter value is increasing, this can indicate that memory is becoming low. Cache\Lazy Write Pages are a subset of Data Flush Pages.

**Cache\Read Aheads/sec** Use this counter to monitor the rate at which the Cache Manager detects that the file is being accessed sequentially. Sequential file access is a very efficient strategy in most cases. During sequential file access, the Cache Manager can read larger blocks of data into the cache on each I/O, thereby reducing the overhead per access.

**Memory\Cache Bytes** Monitor this counter for growth or shrinking of the cache. The value includes not only the size of the cache but also the size of the paged pool and the amount of pageable driver and kernel code. Essentially, these values measure the system's working set.

**Memory\Cache Faults/sec** Observe this counter for the rate at which pages sought in the cache were not found there and had to be obtained elsewhere in memory or on the disk. Compare this counter with Memory\Page Faults/sec and Pages Input/sec to determine the number of hard page faults, if any.

---

**Note** Performance Meter (Perfmtr.exe), a tool on the *Windows 2000 Resource Kit* companion CD and the SDK, lists among other statistics, data about the file system cache. For more information, see the *Windows 2000 Resource Kit* Tools Help (W2rktools.chm) or the SDK documentation.

---

## Interpreting Changes in Cache Counter Values

Administrators need to watch for signs from the cache counters that low memory is causing insufficient cache size and consequently that cache usage is resulting in unnecessary disk I/O, which degrades performance. Monitor for the following conditions, using the counters as described. (Options for improving these conditions include defragmenting the disk and adding memory.)

### Reduction in Cache Size

When memory becomes scarce and working sets are trimmed, the cache is trimmed as well. If the cache grows too small, cache-sensitive processes are slowed by disk operations. To monitor cache size, use the Memory\Cache Bytes and Memory\Available Bytes counters. Note that the effect of a smaller cache on applications and file operations depends on how often and how effectively applications use the cache.

### Frequent Cache Flushing

Frequent cache flushing might occur if data is written to the disk frequently in order to free pages. Data flush counters reflect cache output. Monitor the Cache\Data Flushes/sec and Cache\Lazy Writes Flushes/sec counters.

Cache\Pin Reads/sec tells you how often the cache is reading data with the intention of writing it. A pin read occurs when data is mapped into the cache just to be changed and is then written back to disk. It is pinned in the cache to be sure the data being changed is not written to disk until the update is complete. Creating a file or changing its attributes results in a pin read. Changing file contents does not result in a pin read. Cache\Pin Reads/sec can predict flushing activity.

## High Rates of Cache Misses

High cache miss rates indicate that requested data is not available in physical memory and must be retrieved from the disk. Monitor the Cache object counters that record hit percentages and hit rates (such as Cache\Copy Read Hits % and Cache\Copy Read Hits/sec).

Make sure to review both hit percentage and hit activity counters to ensure that your data gives an accurate picture of cache efficiency. For example, you might notice that your hit percentage is very high (90 percent on average), implying that the cache is highly efficient. However, when you examine hit activity, you might notice that only a few copy reads have occurred during the sample. In this case, relying on the percentages would have given you a false impression of cache effectiveness. Therefore, examining hit percentages over a range of activity rates is recommended.

For computers running Windows 2000 Server, you might also want to monitor MDL Read Hits % and MDL Reads/sec. The Server service might use MDL reads rather than copy reads for large data transfers. MDL reads rely on direct memory access (DMA) for better overall performance.

## High Rates of Cache Faults

Although you cannot tell from the Memory\Cache Faults/sec counter value alone, an increase in cache faults might mean that hard page faults are occurring, resulting in disk activity. Memory\Cache Faults/sec reports fault activity for the system working set, including hard and soft page faults. Because the cache counters do not provide conclusive information about this, you must monitor Memory\Pages Input/sec to determine if hard page faults are occurring.

If Cache Faults/sec values increase and hit-percentage counter values decrease, it is likely there is not enough memory to create an efficient cache. When insufficient memory forces the system to maintain a small cache size, disk I/O tends to increase.

You can see the impact of cache and memory activity on the disk by adding Memory\Pages Output/sec to Cache\Data Flush Pages/sec. This total is approximately equal to the page equivalent of Disk Write Bytes/sec for the PhysicalDisk counters. As the Pages Output/sec and Data Flush Pages/sec counter values increase, so does the value of Disk Write Bytes/sec, reflecting the increasing disk writing activity. (Differences in values can be attributed to the sampling behavior of System Monitor.) Similarly, there is a corresponding read impact: look at Memory\Pages Input/sec; this should equal the number of bytes of memory reported under Disk Read Bytes/sec but expressed in pages for the PhysicalDisk counters as described in "Assessing the Effect of Page Faults on the Disk" earlier in this chapter.

# Resolving Memory and Cache Bottlenecks

While monitoring your memory and cache resources, you might discover memory and cache bottlenecks in your system. As a result, you might need to optimize cache usage, replace applications, or even add new memory. The following tips can assist you with your decision-making process.

## Optimizing Cache Usage

Even though you cannot change the cache itself, there are a few things you can do to make the most of the cache.

- For applications that you develop or maintain, improve the locality of reference in your application's data structures. This improves its cache performance and minimizes its working set so that it uses less space in memory. In addition, it reduces disk access that can slow performance systemwide.
- If you are running Windows 2000 Server, you can direct the Virtual Memory Manager to give the cache higher priority for space than the working sets of processes. See "To adjust Server Optimization settings" earlier in this chapter to change memory management settings. To favor the cache, choose **Maximize Data Throughput for File Sharing**. To favor working sets, choose **Maximize Data Throughput for Network Applications**.
- Change the way work is distributed among workstations. Try dedicating a single computer to memory-intensive applications such as CAD/CAM and large database processors.
- Add memory. When memory is scarce, the cache is diminished and cannot do its job. After the new memory is installed, the Virtual Memory Manager expands the cache to use the new memory.

## Resolving a Memory Bottleneck

Although adding memory is the easy solution to a memory bottleneck, it should not be the first solution you undertake. Try the following, more cost-effective alternatives before spending money on additional memory. If you ultimately decide to add memory, note that the maximum amount of memory recommended for a computer running Windows 2000 Server is 4 GB.

- Monitor your applications and replace or correct those that leak memory or use it inefficiently.
- Modify your application to improve the locality of reference. The Working Set Tuner, included in the Platform SDK, recommends an optimal organization of code functions. For more information, see the MSDN link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

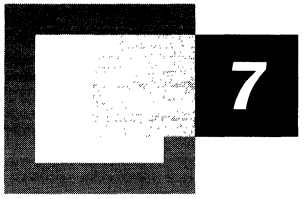
- Increase the size of the paging file. In general, the bigger you can make it, the better it is. You can also have multiple paging files, though you should only have multiple paging files per physical drive when the drive is not partitioned into logical drives. Striped volumes can be used. This improves the read-write rates for the paging file because the work is distributed over multiple disks.
- Check the available space on your disks. If you are using a large paging file and space is not available, this can produce the symptoms of a memory bottleneck.
- To conserve memory, avoid using some display and sound features. Features that can drain memory include animated cursors, desktop icons, large-bitmap wallpaper, and some screen saver programs; removing or disabling them can offer some benefit on a memory-constrained system. Reducing display color depth and screen resolution can also save memory but to a lesser degree.
- To free memory, turn off services you do not use. Stopping services that you do not use regularly saves memory and improves system performance. However, make sure you understand the ramifications of stopping a service before you do so.
- Remove unnecessary protocols and drivers. Even idle protocols use space in the paged and nonpaged memory pools. Drivers also use memory. You can see how a driver uses memory with Pool Monitor (Poolmon.exe).
- Replace 16-bit systems with 32-bit systems for better reliability and performance.
- If you have other computers that are underutilized, move memory-intensive applications to those computers.

# Additional Resources

For more information about developer tools, see the MSDN link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

CHAPTER 7

# Analyzing Processor Activity




After memory consumption, processor activity is the most important thing to monitor in your system. A busy processor might be efficiently handling all the work on your computer, or it might be overwhelmed. Examine processor activity to tell the difference. Use performance counters and *Microsoft® Windows® 2000 Resource Kit* tools to measure processing activity and to determine how to improve performance if necessary.

### In This Chapter

Overview of Processor Monitoring and Analysis 327
Establishing a Baseline for Processor Performance 330
Recognizing a Processor Bottleneck 332
Processes in a Bottleneck 340
Threads in a Bottleneck 345
Advanced Topic: Changing Thread Priority to Improve Performance 351
Eliminating a Processor Bottleneck 362

### Related Information in the Resource Kit

- For information about using System Monitor graphs and counter logs, see "Overview of Performance Monitoring" in this book.
- For information about implementing and optimizing multiprocessor systems, see "Measuring Multiprocessor System Activity" in this book.

# Overview of Processor Monitoring and Analysis

Uniprocessor monitoring and analysis involve many variables. The following steps summarize in-depth monitoring and analysis of processor activity.

- Establish a baseline for processor performance that is reflective of your system's typical workload. Perform this step to characterize your system's workload and identify how applications use the system.
- Examine overall processor usage by viewing processor queue length and processor utilization (also referred to as processor time). Perform this step to obtain an overview of how heavily system resources are used.
- Examine activity that adds to the processing load, such as high rates of interrupts and context switches. Perform this step to determine the efficiency of your system.
- Examine individual processes and their percentage of the overall processor time. Perform this step to learn about the processes on your system.
- Examine the threads—the units of work that make up a process—for each individual process and each thread's processor usage. Perform this step to learn about thread utilization on the system.
- Evaluate thread priorities and change them to see if this provides better performance. (Microsoft does not recommend this is as a long-term solution, but suggests it for testing purposes.) Perform this step to learn how threads interact and to determine which threads are preempting other, lower-priority threads.

Figure 7.1 illustrates the role of processor monitoring in overall system monitoring.

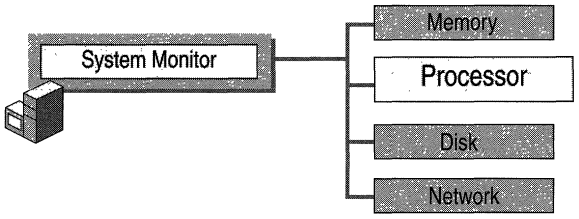



**Figure 7.1 Role of Processor Monitoring in Overall Monitoring Sequence**

Before you begin the monitoring process, become familiar with the counters designed to measure processor activity and the *Windows 2000 Resource Kit* tools that can give you more information about processor workload and performance. The following sections summarize these counters and tools.

# Processor Counters

The System, Processor, Process, and Thread objects contain counters that provide useful information about the work of your processor. Examine the following counters for details about computer processes.

**Table 7.1 Processor Counters**

| Object | Counter | Description |
|---|---|---|
| System | Context Switches/sec | The average rate per second at which context switches among threads on the computer. High activity rates can result from inefficient hardware or poorly designed applications. Compare these counters with Processor\% Privileged Time, Processor\% User Time, and Processor\% Interrupt Time. See "Monitoring Interrupts" and "Monitoring Context Switches" later in this chapter. |
| Processor | Interrupts/sec | The average rate per second at which the processor handles interrupts from applications or hardware devices. High activity rates can indicate hardware problems. Compare these counters with Processor\% Privileged Time, Processor\% User Time, and Processor\% Interrupt Time. See "Monitoring Interrupts" and "Monitoring Context Switches" later in this chapter. |
| System | Processor Queue Length | An instantaneous count of threads that are in the processor queue. See "Observing Processor Queue Length" later in this chapter. |
| Processor | % Processor Time | The percentage of time the processor was busy during the sampling interval. This counter is equivalent to Task Manager's CPU Usage counter. See "Examining the Processor Time Counter" later in this chapter.<br>For the value of total processor utilization systemwide, use the Processor(_Total)\% Processor Time counter. |
| Process | % Privileged Time | The percentage of time a process was running in privileged mode. See "Processes in a Bottleneck" later in this chapter. |
| Process | % Processor Time | The percentage of time the processor was busy servicing a specific process. |
| Process | % User Time | The percentage of time a process was running in user mode. |
| Process | Priority Base | Windows 2000 schedules threads of a process to run according to their priority. Threads inherit base priority from their parent processes. The base priority level of the process can range from lowest to highest: Idle, Normal, High, or Real Time. |
| Thread | Thread State | A numeric value indicating the execution state of the thread.<br>The system numbers threads from 0 through 5; the states seen most often are 1 for ready, 2 for running, and 5 for waiting. Threads with a state of 1 are in the processor queue. |

*continued*

**Table 7.1 Processor Counters** ***(continued)***

| Object | Counter | Description |
|---|---|---|
| Thread | Priority Base | The base priority level (from 1 through 31) for the thread based on the priority class of the process. Windows 2000 schedules threads of a process to run according to their priority. Threads inherit base priority from their parent processes. |
| Thread | Priority Current | The current priority level of a thread. This level can vary during operation. |
| Thread | Context Switches/sec | The average rate per second at which the processor switches context among threads. A high rate can indicate that many threads are contending for processor time. See "Threads in a Bottleneck" later in this chapter. |
| Thread | % Privileged Time | The percentage of time a thread was running in privileged mode. |
| Thread | % User Time | The percentage of time a thread was running in user mode. |

**Note** Because System Monitor samples processor time, the values for processor time counters reported by the Processor, Process, and Thread objects might underestimate or overestimate activity on your system that occurs before or after collection of the sample.

In addition to the preceding list of objects and counters, the Job Object and Job Object Details objects provide information about processor usage. These performance objects are installed by default for monitoring job object performance. The job object makes it possible for developers to manage groups of processes by their processor usage and other factors. For example, job objects make it possible for applications to restrict the amount of processor time a process consumes; this is called process throttling. Process throttling is useful in Web-based administration applications for limiting the amount of processor capacity a site uses over a defined interval, thus avoiding bottlenecks and freeing processor capacity for other tasks. You might also use the job object to manage sharing of CPU time among groups of jobs. In addition to supporting process throttling, job objects help developers control the active number of processes, process identifiers (IDs), priority classes, and processor affinity. For information about creating applications using the job object, see the Platform Software Development Kit (SDK) link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources. For a discussion of application support and job objects, see Getting Started in Windows 2000 Help.

## Resource Kit Tools for Processor Monitoring

The Resource Kit companion CD contains utilities to help you understand and experiment with processor performance, as listed in Table 7.2.

**Table 7.2 Performance Utilities**

| Utility name | Description |
|---|---|
| CPUStres | Simulates processor workload. |
| Qslice | Provides a graphical display of processor usage by process. |

# Establishing a Baseline for Processor Performance

Begin your monitoring routine with an examination of processor usage under your normal workload. By doing so, you can begin to establish a baseline or reference point for processor usage. The baseline is generally not a single value, but a range within which processor usage can fluctuate and still provide acceptable performance. You can use the baseline to identify trends, such as increasing processor demands over time, or to recognize problems that arise from a sudden change.

## Selecting Counters for Baseline Monitoring

To determine the baseline, use the following counters to create logs of processor usage over an extended period (from several weeks to a month).

- Processor\% Processor Time
- System\Processor Queue Length

Be aware of the Idle process when monitoring processor usage. The Idle process runs a thread on each processor. This thread runs when the system is not already running the thread of an active user or system process. System Monitor and Task Manager both use the Idle process to calculate time when the processor is not busy. You can see processor time for the Idle process on the **Processes** tab in Task Manager (called the System Idle Process) or by tracking the Process(Idle)\% Processor Time counter in System Monitor. Notice that the Total instance for this counter includes processor time for the Idle process. To measure the Idle process, use the Process(Idle)\% Processor Time counter, or use the **Processes** tab in Task Manager. Zero idle time could mean that the processor is handling a lot of work, but it could also mean that the processor or central processing unit (CPU) is overloaded.

# Selecting Times for Baseline Monitoring

To monitor processor activity, log the counters of the System, Processor, Process, Thread, PhysicalDisk, and Memory counters for at least several days at an update interval ranging from 15 minutes to an hour. (Use much shorter intervals for bottleneck detection.) Include network counters such as Bytes Total/sec (on the Network Interface object) if you suspect that network traffic might be interrupting the processor too frequently. Because excessive demand on memory and disk resources can cause bottlenecks that appear to affect your processor's performance, also include disk and memory counters in your monitoring configuration to help you determine the true source of any processor bottleneck.

If any applications running on the computer have counters, use these counters to monitor their activity and monitor these values along with system counter values.

Track the values reported at various times of day—for example, while users are logging on or off, while backups are being done, and so forth. As you are monitoring values for these counters, you might see occasional spikes. Typically, you can exclude these from your baseline; the range of values that seem to appear consistently are the ones that constitute your baseline.

---

**Note** Keep in mind application overhead and disk-space usage when you set your monitoring frequency. Frequent updating demands more work and more file storage capacity from your computer. In general, you should experiment with different update intervals to balance these considerations against the level of detail you require for your monitoring data. For more information about how to monitor performance, see Windows 2000 Professional Help.

---

The longer you log, the more accurate your baseline will be. Processor use might be a problem only at certain times of the day, week, or month, and you are likely to see patterns in your workload that are correlated with changes in processor activity if you log for a longer duration. You can even use the log service to schedule monitoring at critical times to determine whether your processor is operating efficiently. During these critical times, you might want to log over intervals as short as every two seconds to get an accurate picture of processor usage on your system. This will help you isolate those applications that heavily stress your processor for further investigation and monitoring.

# Recognizing a Processor Bottleneck

Processor bottlenecks occur when the processor is so busy that it cannot respond to requests for time. Although a high rate of processor activity might indicate an excessively busy processor, a long, sustained processor queue is a more certain indicator. As you monitor processor and related counters, you can recognize a developing bottleneck by the following conditions:

- Processor\% Processor Time often exceeds 80 percent.
- System\Processor Queue Length is often greater than 2 on a single-processor system.
- Unusually high values appear for the Processor(_Total)\Interrupts/sec or System\Context Switches/sec counters.

The most common causes for processor bottlenecks are insufficient memory or excessive numbers of interrupts from disk or network input/output (I/O). To investigate these possible causes, see the following chapters:

- "Evaluating Memory and Cache Usage" in this book
- "Examining and Tuning Disk Performance" in this book
- "Monitoring Network Performance" in this book

Also, the Processor(_Total)\Interrupts/sec counter value might rise dramatically if you've recently added many new applications or users. During periods of low activity the only source of interrupts might be the processor's timer ticks; these are periodic events that increment a processor hardware timer. These occur approximately every 10 to 15 milliseconds, or about 66 to 100 interrupts per second. Interrupt rates vary depending on system workload, including network packets per second and disk I/O operations per second. Watch for interrupt values that fall out of a normal range (expect these to be in the 1000s of interrupts on Microsoft® Windows® 2000 Server and from 200 to 300 on Microsoft® Windows® 2000 Professional). If Processor\% Interrupt Time exceeds 20 to 30 percent per processor, it might indicate that the system is generating more processor interrupts than it can handle. If this is the case, you might need to upgrade some of your components. For more information, see "Monitoring Network Performance" in this book.

If a processor bottleneck does not exist but you are dissatisfied with system performance, and you have ruled out memory and other hardware factors, consider the following options to improve CPU response time or throughput:

- Schedule processor-intensive applications to run when the system load is low. Use **Scheduled Tasks** in Control Panel or the **at** command to do this.

- Upgrade to a faster processor. Upgrading to a higher-speed processor with a larger L2 cache will expedite processing regardless of your system's workload.

  When upgrading to a faster processor, check with the chip vendor to ensure that you use the correct memory speed for the chip. Incompatible memory speed could cause a computer with a faster processor to appear to run more slowly than a computer with a slower processor.

---

**Note** Using multiple processors rather than switching to a faster one might not automatically improve performance in a dramatic manner. For example, a 200-megahertz (MHz) dual-processor computer might not perform equally to a 400-MHz uniprocessor computer with all workloads because of overhead inherent in synchronization. Because scaling can incur some overhead, it is important to be aware of the factors involved and how to manage them. For more information, see "Measuring Multiprocessor System Activity" in this book.

---

If conditions do not warrant immediate processor replacement, begin monitoring processor activity and system performance as described in the following sections.

## Examining the Processor Time Counter

The Processor\% Processor Time counter determines the percentage of time the processor is busy by measuring the percentage of time the thread of the Idle process is running and then subtracting that from 100 percent. This measurement is the amount of processor utilization. Although you might sometimes see high values for the Processor\% Processor Time counter (70 percent or greater depending on your workload and environment), it might not indicate a problem; you need more data to understand this activity. For example, high processor-time values typically occur when you are starting a new process and should not cause concern.

---

**Note** The value that characterizes high processor utilization depends greatly on your system and workload. This chapter describes 70 percent as a typical threshold value; however, you may define your target maximum utilization at a higher or lower value. If so, substitute that target value for 70 percent in the examples provided in this section.

---

To illustrate, consider that Windows 2000 allows an application to consume all available processor time if no other thread is waiting. As a result, System Monitor shows processor-time rates of 100 percent. If the threads have equal or greater priority, as soon as another thread requests processor time, the thread that was consuming 100 percent of CPU time yields control so that the requesting thread can run, causing processor time to lessen. For a discussion of priority and scheduling, see "Threads in a Bottleneck" later in this chapter.

If you establish that processor-time values are consistently high during certain processes, you need to determine whether a processor bottleneck exists by examining processor queue length data. Unless you already know the characteristics of the applications running on the system, upgrading or adding processors at this point would be a premature response to persistently high processor values, even values of 90 percent or higher. First, you need to know whether processor load is keeping important work from being done. You have several options for addressing processor bottlenecks, but you need to first verify their existence.

If you begin to see values of 70 percent or more for the Processor\% Processor Time counter, investigate your processor's activity further, as follows:

- Examine System\Processor Queue Length.
- Identify the processes that are running when Processor\% Processor Time and System\Processor Queue Length values are highest.

## Observing Processor Queue Length

A collection of one or more threads that is ready but not able to run on the processor due to another active thread that is currently running is called the processor queue. The clearest symptom of a processor bottleneck is a sustained or recurring queue of more than two threads. Although queues are most likely to develop when the processor is very busy, they can develop when utilization is well below 90 percent. This can happen if requests for processor time arrive randomly and if threads demand irregular amounts of time from the processor. For more information about monitoring and adjusting thread scheduling, see "Threads in a Bottleneck" later in this chapter.

The System\Processor Queue Length counter shows how many threads are ready in the processor queue, but not currently able to use the processor. Figure 7.2 shows a sustained processor queue with utilization ranging from 60 to 90 percent. Notice that the default scale for the Processor Queue Length counter value is 10. Therefore, System Monitor graphs a queue that is two threads long as 20. You can change the scale factor using the **Data** properties tab in System Monitor.

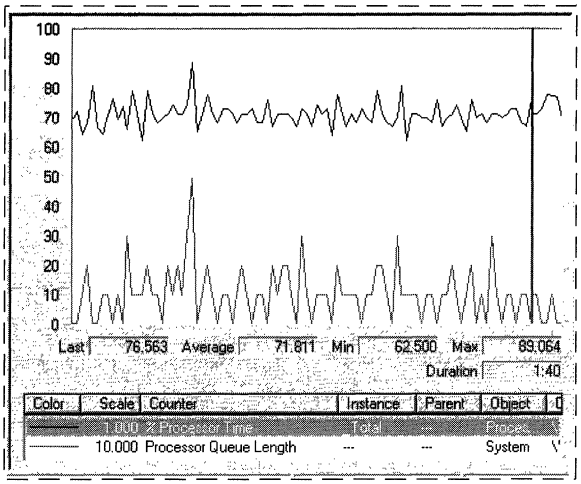



**Figure 7.2 Sustained Processor Queue with Rising Processor Usage**

In Figure 7.2, the line across the top represents Processor(_Total)\% Processor Time. The lower line is System\Processor Queue Length.

Figure 7.3 shows a sustained processor queue accompanied by processor use at or near 100 percent.

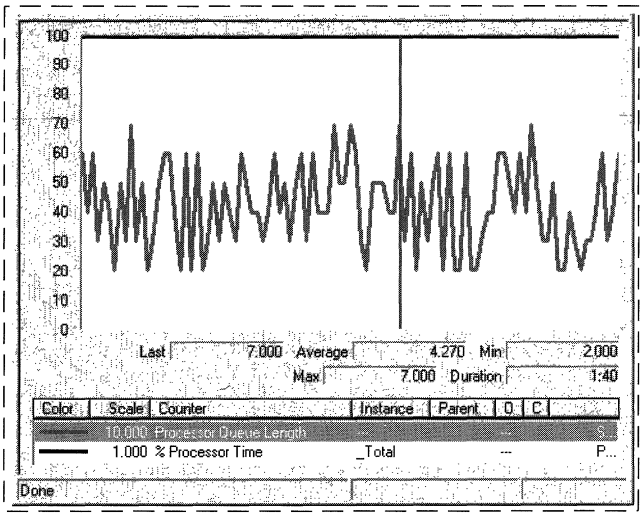



**Figure 7.3 Sustained Processor Queue with Maximum Processor Usage**

Figure 7.4 illustrates how a processor bottleneck interferes with your computer's performance. It shows that when a processor is already at 100 percent utilization, starting another process does not accomplish more work.

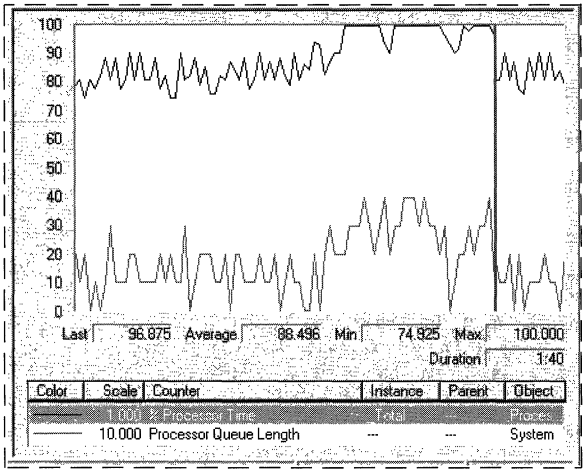



**Figure 7.4 Saturated Processor**

In Figure 7.4, the dark line running across the top of the graph is Processor(_Total)\% Processor Time. The line below it is System\Processor Queue Length. Midway through the sample interval, a process with three threads was started. The graph illustrates that the queue increased as a result of this added workload. Some of the threads of the added process might be in the queue, or they might be running, having displaced the threads of a lower-priority process. Nonetheless, because the processor was already at maximum capacity, it can accomplish no additional work.

If your system's counter values appear similar to those in Figure 7.4, this indicates a bottleneck. Logging, over time, will reveal any patterns associated with the bottleneck. For example, you might find that bottlenecks occur when certain processes are running or at a certain time of day. In this case, you might be able to eliminate the bottleneck by balancing the workload between computers—that is, running the process on another, less loaded computer.

However, if sustained queues appear frequently, you need to investigate the processes that are running when threads collect in the queue. To determine this:

- Identify the processes that are consuming processor time. Determine whether a single process or multiple processes are active during a bottleneck. Running processes appear in the **Instance** box when you select the Process\% Processor Time counter. For more information, see "Processes in a Bottleneck" later in this chapter.
- Scrutinize the processor-intensive processes. Determine how many threads run in the process and watch the patterns of thread activity during a bottleneck.

- Evaluate the priorities at which the process and its threads run. You might be able to eliminate a bottleneck merely by adjusting the base priority of the process or the current priorities of its threads. However, Microsoft does not recommend this as a long-term solution. Use Task Manager to find the base priority of the process.

---

**Note** Different guidelines apply for queue lengths on multiprocessor systems. For busy systems (those having processor utilization in the 80 to 90 percent range) that use thread scheduling, the queue length should range from one to three threads per processor. For example, on a four-processor system, the expected range of processor queue length on a system with high CPU activity is 4 to 12.

On systems with lower CPU utilization, the processor queue length is typically 0 or 1.

---

There are other objects that track processor queue length. The Server Work Queues\Queue Length counter reports the number of requests in the queue for the processor on the selected server. For more information about monitoring the Server Work Queues object, see "Monitoring Network Performance" in this book.

## Monitoring Interrupts

Sharply rising counts for interrupts can affect your processor's performance, and you should investigate their cause. The Processor\Interrupts/sec counter reports the number of interrupts the processor is servicing from applications or hardware devices. You should expect interrupts to range upward from 1,000 per second for computers running Windows 2000 Server and upward from 100 per second for computers running Windows 2000 Professional. This interrupt rate is dependent on the rate of disk I/O operations per second and network packets per second. If your interrupt counter values are out of range, there might be hardware problems such as a conflict between the hard-disk controller and a network adapter. You can use System Information and Device Manager in the Computer Management console to check for problems with the disk controller or network adapter.

You might want to monitor interrupts along with I/O activity involving both disks and network cards. Use the Disk Reads/sec or Disk Writes/sec counters on the PhysicalDisk object to monitor disk I/O as described in "Examining and Tuning Disk Performance" in this book. Use the network transmission counters to monitor network activity as described in "Monitoring Network Performance" in this book. You can tell if interrupt activity is becoming a problem by determining the ratio of interrupts to I/O operations. An optimal ratio is one interrupt to four or five I/O operations. A one-to-one correspondence between these factors indicates poor performance and requires action.

If network or disk I/O is involved, you should consider upgrading to a controller and a driver that support interrupt moderation or interrupt avoidance. Interrupt moderation allows a processor to process interrupts more efficiently by grouping several interrupts to a single hardware interrupt. Interrupt avoidance allows a processor to continue processing interrupts without new interrupts being queued until all pending interrupts are complete. For more information about managing interrupts from network adapters, see "Monitoring Network Performance" in this book.

High values for % Processor Time for threads of the System process can also indicate a problem with a device driver.

## Monitoring Context Switches

A context switch occurs when the kernel switches the processor from one thread to another—for example, when a thread with a higher priority than the running thread becomes ready. Context switching activity is important for several reasons. A program that monopolizes the processor lowers the rate of context switches because it does not allow much processor time for the other processes' threads. A high rate of context switching means that the processor is being shared repeatedly—for example, by many threads of equal priority. A high context-switch rate often indicates that there are too many threads competing for the processors on the system.

---

**Note** The rate of context switches can also affect performance of multiprocessor computers. For information about how to monitor and tune context-switch activity on multiprocessor systems, see "Measuring Multiprocessor System Activity" in this book.

---

You can view context switch data in two ways:

- The System\Context Switches/sec counter in System Monitor reports systemwide context switches.
- The Thread(_Total)\Context Switches/sec counter reports the total number of context switches generated per second by all threads.

Although these counters might vary slightly due to sampling, generally they will be nearly equal.

Figure 7.5 plots System\Context Switches/sec during a transient bottleneck.

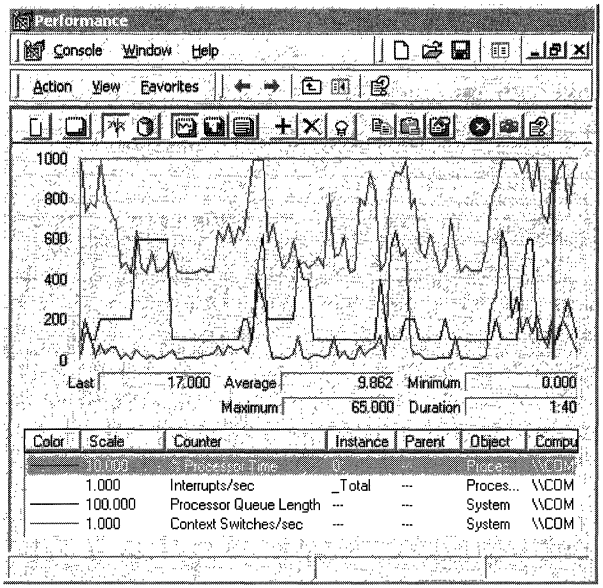



**Figure 7.5 Systemwide Context Switches During a Processor Bottleneck**

In Figure 7.5, Processor(_Total)\% Processor Time jumps to about 60 percent during the sample interval. System\Processor Queue Length (scaled by a factor of 10), shows that the queue varies from 2 to 6, with a mean near 4. System\Context Switches (shown scaled by a factor of 10), reveals an average of about 750 switches per second. A rate of context switches from 500 to 2,000 per second might indicate a problem with a network adapter or a device driver or that you are using an inefficient server-based application that spawns too many threads.

The Pviewer utility on the on the Windows 2000 operating system CDreports context switch data.For information about installing and using the Windows 2000 Support Tools and Support Tools Help, see the file Sreadme.doc in the \Support\Tools folder of the Windows 2000 operating system CD.

# Processes in a Bottleneck

After you have identified a processor bottleneck, you next need to determine whether a single process is using the processor or whether the processor is being consumed by running many processes. To do this, log processor time used by each process running on your computer, as follows:

- Select the Process object.
- Select the % Processor Time counter.
- Select each process instance.

---

**Important** All processes that are running will appear in the **Instance** box listed by the name of the associated executable program (for example, Windows Explorer appears as "explorer" in the **Instance** box). Note that if you are running multiple instances of the same executable program, System Monitor lists these under the identical name; thus, you will need to track these by their process identifiers. You can find the process identifier by using the Process\Process ID counter or by adding the PID column in Task Manager. For more information, see "Threads in a Bottleneck" later in this chapter.

---

For more information about MS-DOS-based (shown as NTVDM for Windows NT Virtual DOS Machine) and Win16-based processes that appear differently in the user interface. See "Overview of Performance Monitoring" in this book.

## Identifying Active Processes

To determine the processing load generated by your typical workload, include the instances for the processes that you normally run. If you find that you don't normally use some processes (such as background services) that are already running, stop these processes and measure the impact on your processing load. This might be an easy way to improve your computer's processing efficiency. However, before doing so, make sure you understand the possible effects. To stop a service, complete the following procedure.

▶ **To stop a service**

1. Click **Start**, point to **Programs**, and then click **Administrative Tools**.
2. Click **Computer Management**, and then, in the console that appears, click **Services and Applications**.
3. Under **Services and Applications**, double-click **Services**.
4. Right-click the name of a service, and then click **Stop**.

# Isolating Processor-Intensive Workloads

If the threads of a process are using a high percentage of CPU time, the process should be analyzed to determine if the application's performance can be optimized. For information about improving application performance, see the Platform Software Development Kit (SDK) link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources. If optimization does not yield satisfactory results, you need to add processor resources.

Device driver problems can cause high % Processor Time values for the System process. In addition, if you have screen savers running on your system, note that they use a large amount of processor time, particularly ones that use OpenGL (a software interface that supports production of high-quality, three-dimensional color imaging). Monitor your screen saver executable and, if necessary, change to one that uses less processor time. (Typically this is only applicable to servers.)

If a single process is using the processor, the chart line associated with that process will be the highest one in the graph, such as in Figure 7.6, which is an actual histogram of a processor bottleneck caused by a single process. Running CPU Stress, a utility on the *Windows 2000 Resource Kit* companion CD, produced the results shown in this example.

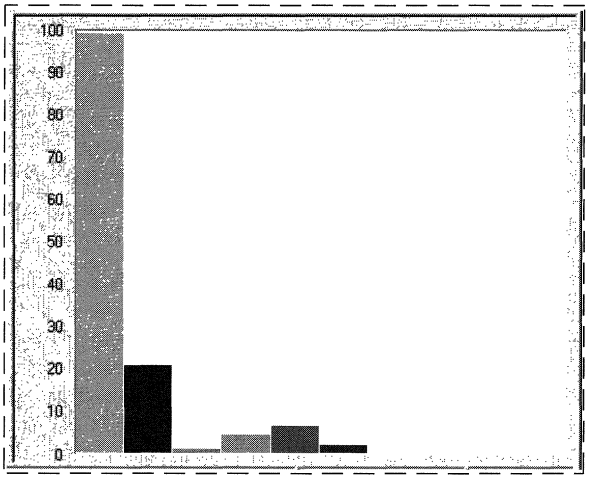



**Figure 7.6 Processor Bottleneck Caused by a Single Process**

This histogram shows that a single process (represented by the tallest bar) is highly active during a bottleneck; its threads are running for more than 90 percent of the sample interval. If this pattern persists and a long queue develops, it is reasonable to suspect that the application running in the process is causing the bottleneck.

Note that a highly active process is a problem only if a queue is developing. If you are not satisfied with response time and throughput, you can choose to upgrade to a faster processor to achieve better performance.

If you suspect that an application is causing a processor bottleneck, stop using the application for a few days or move it to a different computer. Another option is to schedule the process to run outside of peak operating hours. Then log processor use again. If the problem disappears, it is likely that the application caused it.

## Reducing Single-Process Bottlenecks

If you cannot use another computer and you have access to the application source code, you can tune the application to increase efficiency. Start by using a profiler, an analysis tool that you can use to examine the run-time behavior of your programs. Profiling enables you to analyze how the application is spending processor time. The Platform Software Development Kit (SDK) includes tools and methods for profiling and optimizing applications, including instructions for developing performance counters to monitor the inner workings of your application. To tune an application to be less CPU-intensive, use SDK utilities such as Call Attributed Profiler (CAP) or File I/O Synchronization Win32 API Profiler (FIOSAP). You can also use APIMon and Kernprof, which are included on the *Windows 2000 Resource Kit* companion CD.

If tuning efforts do not reduce the application's load on your processor, or if you do not have access to the application source code, you can:

- Consider adding a processor or upgrading the one you have. If your application is multithreaded, adding a processor can alleviate a bottleneck because multithreaded applications can run on multiple processors. However, single-threaded applications will not benefit because the system cannot distribute their thread activity across processors; these applications need faster processors or need to run on a computer with extra processing capacity. In general, a faster CPU will probably result in a greater performance gain than installing additional processors because the management of the work performed by multiple CPUs also consumes processor time.
- Investigate the activity of threads in the process. For more information about examining thread behavior and changing thread scheduling patterns to ensure that the necessary processes get processor time, see "Threads in a Bottleneck" later in this chapter. This section also contains instructions for determining whether a process is single-threaded or multithreaded; this distinction is important in making an upgrade decision.
- If processor use continues to create a bottleneck even without the application that you first suspected, repeat the preceding steps and carefully monitor the processes that are active when the queues are longest.
- Consider replacing the application with one that has been optimized to run under Windows 2000.

# Observing Processor Consumption by Multiple Processes

Figure 7.7 shows a histogram of processor time for many active processes. This example was produced by running two instances of CPU Stress, which consumes processor cycles at the priority and activity levels you specify.

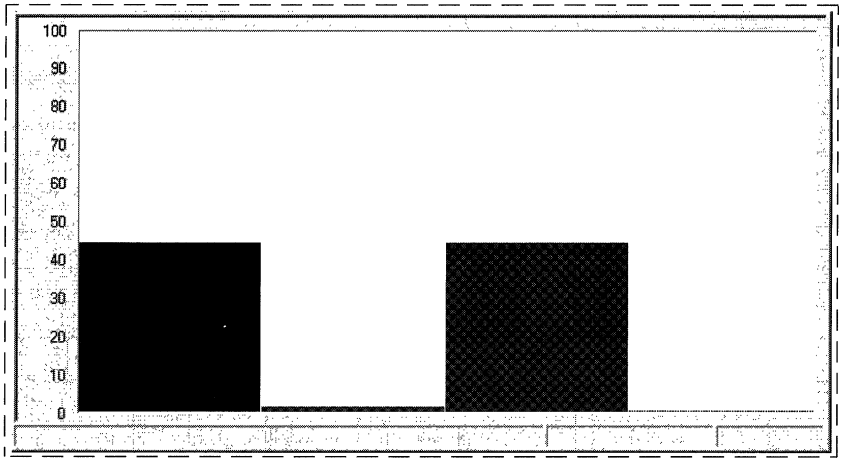



**Figure 7.7 Processor Time for Multiple Active Processes**

In this example, two processes are consuming the processor while sharing it nearly equally. Although each process is using only 45 percent of the processor, the result is the same as a single process using 90 percent of processor time.

Figure 7.8 shows System\Processor Queue Length during this bottleneck.

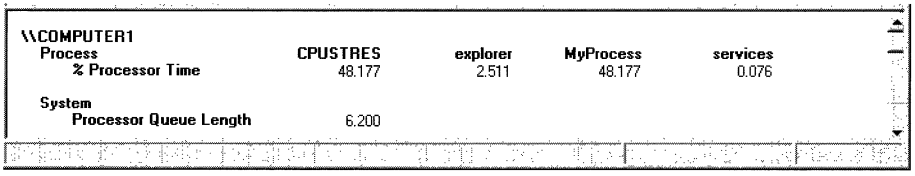



**Figure 7.8 Processor Queue Length During Activity of Many Processes**

In Figure 7.8, Processor\% Processor Time for all processes is close to 100 percent during the sample interval. System\Processor Queue Length reveals a long queue, averaging over six threads.

Figure 7.9 shows Task Manager during the same bottleneck. It shows that two CPU Stress processes are each using about half of the time of the single processor on the computer. (Task Manager shows current values, so you need to watch the display to see changes in processor use for each process.)

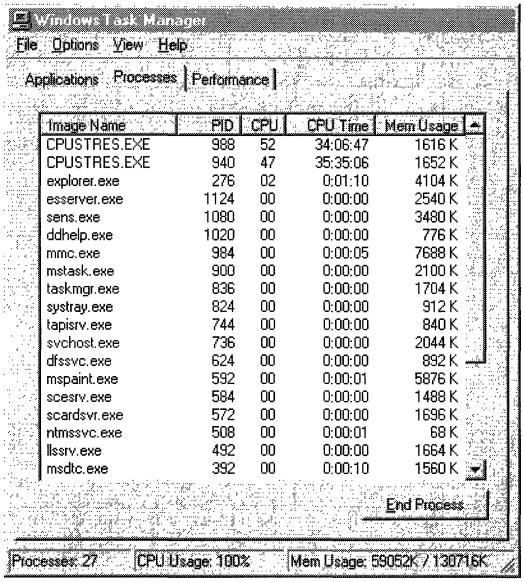



**Figure 7.9 CPU Usage for Multiple Processes in Task Manager**

At this point, you can choose to add a processor or upgrade the one you have, or you can investigate the activity further by researching thread behavior. Although a faster processor might help this situation somewhat, multiple-process bottlenecks are best resolved by adding another processor. Multithreaded processes, including multithreaded Windows 2000 services, benefit the most from additional processors because their threads can run simultaneously on multiple processors. You might want to partition the processes among the processors for optimal efficiency. For more information, see "Measuring Multiprocessor System Activity" in this book.

To find out more about how a particular process uses the processor, examine the Process\% User Time and Process\% Privileged Time counters, followed by Thread object counters, as described in the following section.

For more information about determining if your application is multithreaded, see "Threads in a Bottleneck" later in this chapter. You might also want to find out whether all threads in a multithreaded process are active during bottlenecks. The benefit of adding processors depends on whether you have a lot of active threads. You could find, as a result of monitoring, that threads in a process are inactive most of the time, so adding a processor to handle these inactive threads is a waste of money and capacity.

For more information about the benefits of adding processors to manage a larger workload (called scaling) and how to determine whether scaling is appropriate, see "Measuring Multiprocessor System Activity" in this book.

# Threads in a Bottleneck

Investigate the individual thread or threads of the process or processes running during a bottleneck to understand more about the activity consuming the processor. Monitor the following factors to understand how thread activity is contributing to the problem, whether the cause is a single process or multiple processes:

- The number of threads in each process that is running during a bottleneck
- The amount of processor time a thread is consuming
- The priority level at which threads are scheduled to run
- The amount of time the threads are using the processor in privileged mode

You can use performance counters to analyze thread activity and adjust thread scheduling to allow more processor time for bottlenecked processes.

Apart from adjusting the thread's scheduling priority, you cannot alter thread behavior without changing the program code of the associated application. However, if you have access to application source code, you can write counters to monitor thread activity at a lower level. For more information, see the Platform Software Development Kit (SDK) link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

## Single vs. Multiple Threads in a Bottleneck

Bottlenecks can result from activity of multiple threads in a single process, single threads in multiple processes, or multiple threads in multiple processes. Because these problems require different solutions, you need to first distinguish their causes.

To study threads during a bottleneck, log counters of the Processor, Process, and Thread objects for several days at an update interval of 60 seconds. This will allow you to look at thread activity during typical operating conditions and help you associate that activity with processor usage.

**Important** Performance counter values for threads are subject to error when threads are stopping and starting. Faulty values sometimes appear as large spikes in the data. For more information, see "Overview of Performance Monitoring" in this book.

### ▶ To determine whether a process is single-threaded or multithreaded

1. Right-click the Windows task bar and select **Task Manager**.
2. In **Task Manager**, click the **Processes** tab and, on the **View** menu, click **Select Columns**.
3. In the **Select Columns** dialog box, select the **Thread Count** check box, and then click **OK**.

   This column shows the total number of threads associated with the process.

Figure 7.10 illustrates how Task Manager displays the number of threads running in a process, and the name of the process.

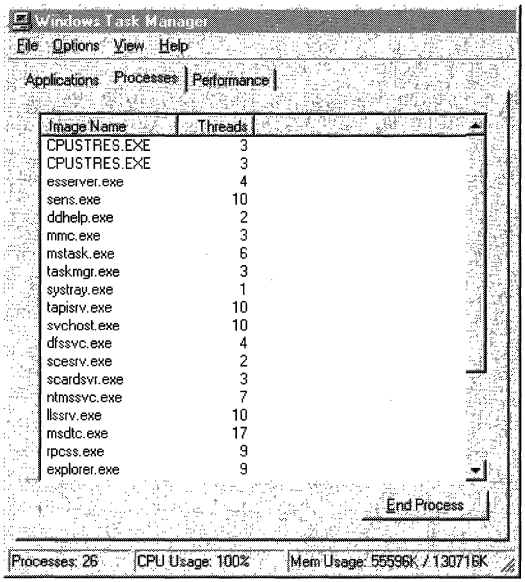



**Figure 7.10 Number of Threads Initiated by a Process Shown in Task Manager**

In System Monitor or Counter Logs, select the Thread object and look at all instances listed in the **Instance** box. If there are several thread identifiers (IDs) listed, then the process is multithreaded. Figure 7.11 illustrates multiple threads of a process as they are listed in the **Instances** box in the System Monitor user interface.

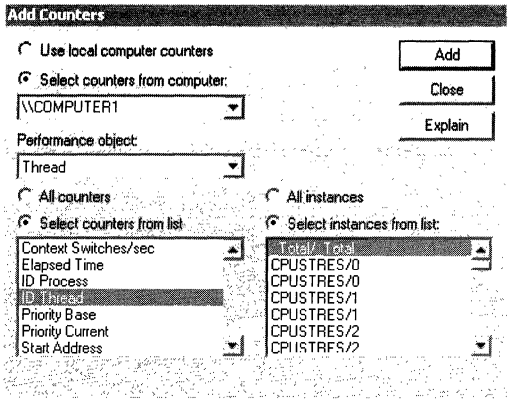



**Figure 7.11 Thread Instances Shown When Adding Counters in System Monitor**

System Monitor identifies threads by process name and thread number. The order in which the threads appear on the chart depends on the order in which you add them to your chart. The thread number shown in the **Instance** box represents the order in which the threads started, and it can change even as the process runs.

Thread identifiers are valid only during the lifetime of the thread; they are recycled when the thread terminates. Thread numbers can change while running, so it is best to monitor by thread identifier. The Tlist utility provides thread identifier information. For information about Tlist, see Windows 2000 Support Tools Help. For information about installing and using the Windows 2000 Support Tools and Support Tools Help, see the file Sreadme.doc in the \Support\Tools folder of the Windows 2000 operating system CD.

If a process is multithreaded, adding a processor will help improve performance. If it is single-threaded, you can improve performance by using a faster processor. These solutions are more advanced and more relevant to developers who also might want to tune the problem applications.

## Charting Processor Usage Per Thread

Observing processor time by threads in a process provides additional information about the activity of the processor during a bottleneck. System Monitor provides the Thread\%Processor Time counter for monitoring processor usage for each thread in a running process. If you have determined that a process is single-threaded, you do not need to track the processor time for the process's thread because this will be nearly identical (except for small variations due to sampling) to the processor time you recorded when tracking the process itself.

Figure 7.12 shows Thread\% Processor Time for all initialized threads during a bottleneck. Each bar of the histogram represents the processor time of a single thread.

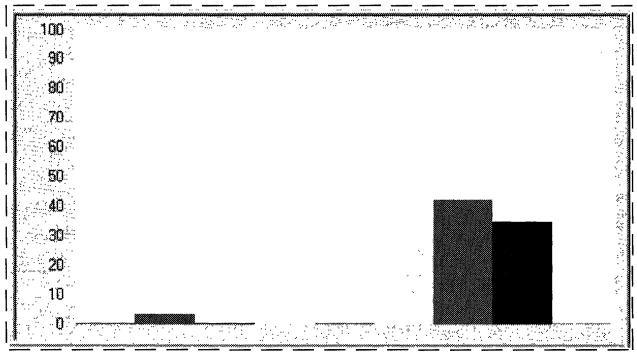



**Figure 7.12 View of Threads and Processor Usage**

Figure 7.12 shows that the three threads of the CPU Stress process are dominating the pattern of processor use, although a few other threads are getting some processor time.

If your thread activity appears similar to the preceding figure and a long queue is developing, some applications on your system are probably not getting enough processor time to run as efficiently as you would like. To investigate the threads of the process and how they use the processor, monitor context switching and user-mode vs. kernel-mode CPU usage, as described in the following sections.

## Context Switches

The Thread\Context Switches/sec counter in System Monitor provides another perspective on how the operating system schedules threads to run on the processor. A context switch occurs when the kernel switches the processor from one thread to another. A context switch might also occur when a thread with a higher priority than the running thread becomes ready or when a running thread must wait for some reason (such as an I/O operation). The Thread\Context Switches/sec counter value increases when the thread gets or loses the time of the processor.

In the course of a context switch, at least two threads are changing their thread state. However, one of the threads may be the idle thread of a given processor. A careful examination of context switch data reveals the patterns of processor use for a thread and indicates how efficiently a thread shares the processor with other threads of the process or other processes.

The System\Context Switches/sec counter that reports systemwide context switches should be close to if not identical to the value provided by the _Total instance of the Thread\Context Switches/sec counter. Monitoring over time should help you determine the range by which the two counters' value might vary.

Interpret the data cautiously. A thread that is heavily usingthe processor lowers the rate of context switches because it does not allow much processor time for other processes' threads. A high rate of context switching means that the processor is being shared repeatedly—for example, by many threads of equal priority. It is a good practice to minimize the context switching rate by reducing the number of active threads on the system. The use of thread pooling, I/O completion ports, and asynchronous I/O can reduce the number of active threads. Consult your in-house developers or application vendors to determine if the applications you are running provide tuning features that include limiting the number of threads.

A context switching rate of 300 per second per processor is a moderate amount; a rate of 1000 per second or more is high. Values at this high level may be a problem.

You can determine whether context switching is excessive by comparing it with the value of Processor\% Privileged Time. If this counter is at 40 percent or more and the context-switching rate is high, then you can investigate the cause for high rates of context switches.

## User Mode and Privileged Mode

You can determine the percentage of time that threads of a process are running in user and privileged mode. User mode is the processing mode in which applications run. Privileged or kernel mode is the processing mode that allows code to have direct access to all hardware and memory in the system. Developers might want to know how much time a process is spending in each mode and what function is using the processor in this way.

I/O operations and other system services run in privileged (kernel) mode; user applications run in user mode. Unless they are graphics-intensive or I/O-intensive (such as file and print services), most applications should not be processing much work in kernel mode.

System Monitor has % Privileged Time and % User Time counters on the System, Processor, Process, and Thread objects. These counters are described in "Processor Counters" earlier in this chapter. System Calls/sec is also a useful indicator of privileged time usage because application calls to the operating system are handled in privileged mode.

In the user time and privileged time counters, System Monitor displays the proportion of total processor time that the process is spending in user or privileged mode.

Figure 7.13 is a System Monitor report on the proportion of user and privileged time for three processes.

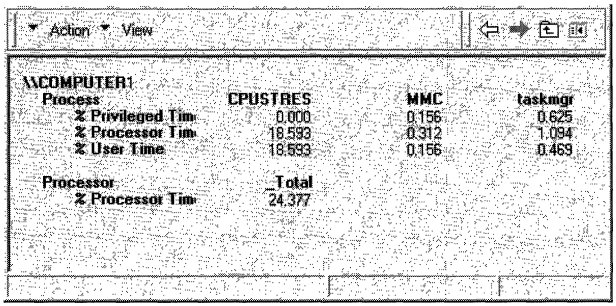



**Figure 7.13 User and Privileged Time for Processes**

In Figure 7.13, Microsoft Management Console (MMC), the process in which System Monitor is running, is running mainly (80 percent) in privileged mode. Taskmgr, the Task Manager process, is also running mainly in privileged mode (70 percent), though this proportion varies significantly as the process runs. In contrast, CpuStres, the process for the CPU Stress test program, runs entirely in user mode all of the time.

Figure 7.14 shows the proportion of user and privileged time for each thread of the Task Manager process.

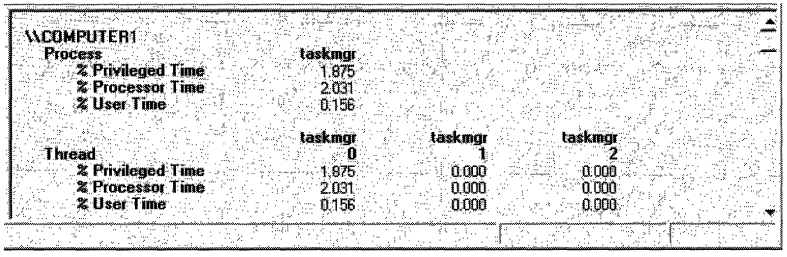



**Figure 7.14 User and Privileged Time for a Process and Its Threads**

The Process Viewer (Pviewer.exe) utility displays the proportion of user and privileged time for each running process and, separately, for each thread in the process. In Process Viewer, the user and privileged mode percentages for each process always total 100 percent because idle time is included. However, in System Monitor the percentages for each process reflect the amount of nonidle processor time actually used in each mode and instead total the amount of nonidle time. Therefore, the value for each process might not total 100 percent. To see the process times add up to 100 percent, combine the percentages for all processes including the Idle process.

For information about Pviewer, see Windows 2000 Support Tools Help. For information about installing and using the Windows 2000 Support Tools and Support Tools Help, see the file Sreadme.doc in the \Support\Tools folder of the Windows 2000 operating system CD.

# Advanced Topic: Changing Thread Priority to Improve Performance

After observing the threads that use the greatest amount of processor time, monitor the dispatch states of the threads. This will tell you which threads are running and which threads are ready. Most important, monitoring thread states on your system can help you identify which threads are piling up in the queue and which threads are actively running at various times.

The Thread\Thread State counter provided by System Monitor reports the current execution state (also known as dispatch state) of a thread. System Monitor reports thread state as a numeric value from 0 through 7, corresponding to whether the thread is ready, running, terminated, and so on.

Table 7.3 lists the thread states you will typically see.

**Table 7.3 Typical Thread States**

| Thread state | Description |
|---|---|
| 0 | Initialized. |
| 1 | Ready. The thread is prepared to run on the next available processor. |
| 2 | Running. |
| 3 | Standby. The thread is about to use the processor. |
| 4 | Terminated. |
| 5 | Waiting. The thread is not ready to run, typically because another operation (for example, involving I/O) must finish before the thread can run. |
| 6 | Transition. The thread is not ready to run because it is waiting for a resource (such as code being paged in from disk). |
| 7 | Unknown. The thread is in an unknown state. |

To determine which threads are contending for the processor, track the states of all threads in the system using System Monitor. Figure 7.15 shows a histogram. The vertical maximum for the chart is set to 10 to make it easier to see the values; an alternative for easier viewing is to display the thread-state values in a report view.

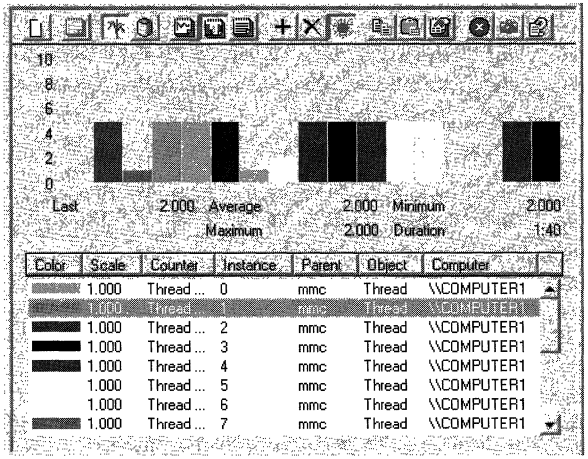



**Figure 7.15 Display of Thread States**

Notice that the preceding figure plots a thread of the MMC process with the steady value of **2** for running. This is the thread of the System Monitor snap-in that is collecting the data that you are monitoring. As long as System Monitor is running, one of its threads will be shown to be running. Other threads' state values alternate between **1** for ready and **5** for waiting.

Plotting thread-state data in a chart rather than a histogram might make it easier to view the switching of thread states. Note in Figure 7.16 how a process's thread moves from the waiting state (plotted on the chart at 5) to the ready state (plotted on the chart at 1).

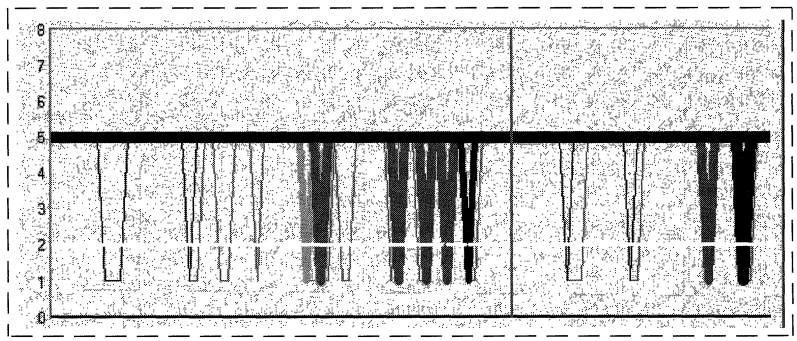



**Figure 7.16 Changing Thread States**

To find out how long threads remain in a particular state, log Thread\Thread State and, when the log is completed, export it to Microsoft Excel. A sample log is shown in Figure 7.17.

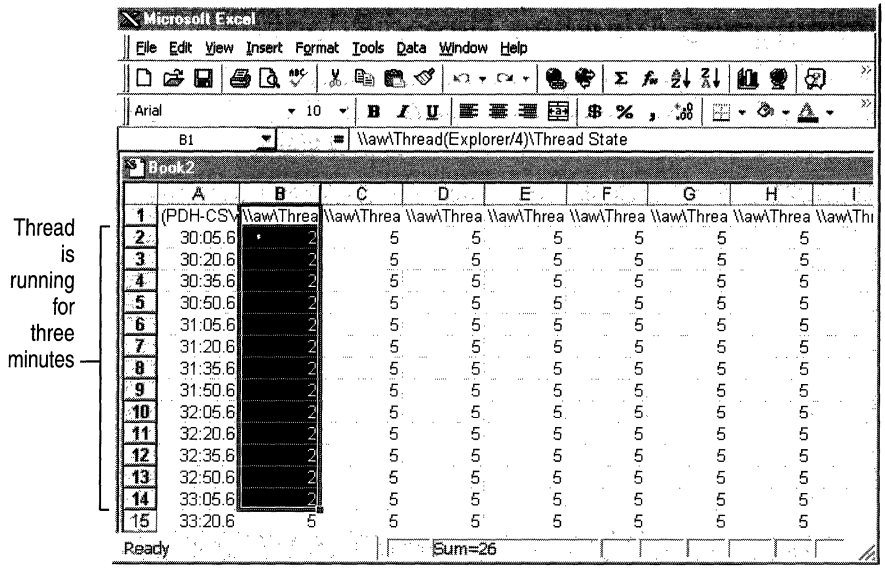



**Figure 7.17 Sample Log Output Viewed in Microsoft Excel**

By looking at log output, you can get an idea of the length of time that a thread remains in a state by totaling the number of seconds that elapsed until the thread's state changed. However, it is important to note that, due to the fact that sampling omits some data, you might not see all of the state changes that occur.

In addition, the Thread\Thread Wait State counter and Perfmon4.exe on the *Windows 2000 Resource Kit* companion CD give you information about why a thread is in a waiting state.

The value reported for Thread Wait Reason is a code. The Counters Help file on the *Windows 2000 Resource Kit* companion CD provides descriptions for these codes.

# Examining and Adjusting Thread Priority

Examining thread context switching and thread state gives you information about when threads in a bottleneck are being scheduled to run by the operating system and when threads are being held in the queue prior to running. Although the operating system is inherently designed to optimize the scheduling of threads, you have some control over this behavior, to adjust for situations in which scheduling behavior on your system is unsatisfactory. This section describes how you can determine a thread's scheduling priority and how you can adjust thread priority to reduce bottlenecks and allow blocked threads to run.

---

**Important** To ensure optimum performance on production systems, Microsoft recommends that you adjust priorities of processes first in a test environment. In addition, you should make these adjustments only if you have an in-depth understanding of priority settings and their effect on other processes and the operating system.

---

## Priority Class and Priority

Under the preemptive multitasking strategy built into Windows 2000, threads and processes are assigned a priority for scheduling purposes. A thread's priority determines the order in which it is scheduled to run on the processor.

A thread's priority is based on the priority class of its parent process. The four process priority classes are:

- **Idle.** Screen savers and other processes that periodically update the display typically use the Idle class.
- **Normal.** The default priority class for a process is Normal.
- **High.** Processes that run in the High priority class receive the majority of processor time.
- **Real Time.** Many kernel-mode system processes, such as those that manage mouse and keyboard input and other device operations, run in the Real Time priority class.

Each process's priority class sets a range of priority values (between 1 and 31, where 1 is lowest and 31 is highest), and the threads of that process have a priority value that is within that range. (Priority 0 is reserved for system use.) If the priority class is Real Time (priorities 16 through 31), the thread's priority cannot change while the thread is running. If you have at least one priority 31 thread running, other threads cannot run.

On the other hand, threads running in all other priority classes are variable, meaning that the thread's priority can change while the thread is running. For threads in the Normal or High priority classes (priorities 1 through 15), the thread's priority can be raised or lowered by up to a value of 2 but cannot fall below its original, program-defined base priority. When the base priority is adjusted to optimize scheduling, the resulting value is called the thread's dynamic priority.

Table 7.4 associates each process priority class with relative thread priorities, ranked from highest priority to lowest. Notice that the highest priority class is Real Time and the lowest is Idle.

**Table 7.4 Process Priority Classes With Relative Thread Priorities**

| | Process priority classes | | | |
|---|---|---|---|---|
| **Thread priorities** | **Real time** | **High** | **Normal** | **Idle** |
| Time critical | 31 | 15 | 15 | 15 |
| Highest | 26 | 15 | 10 | 6 |
| Above normal | 25 | 14 | 9 | 5 |
| Normal | 24 | 13 | 8 | 4 |
| Below normal | 23 | 12 | 7 | 3 |
| Lowest | 22 | 11 | 6 | 2 |
| Idle | 16 | 1 | 1 | 1 |

## Thread Scheduling

The scheduling routines of the operating system run the highest-priority ready thread uninterrupted during a *quantum* (also known as a time slice—the maximum amount of time a thread can run before the system checks for another ready thread of the same priority to run). If a higher-priority thread becomes ready during the quantum, the lower-priority thread is interrupted and the higher-priority thread is run. Otherwise, threads running at the same priority are scheduled in a round-robin fashion, switching among threads in order, and run until the quantum expires.

In general, Windows 2000 always runs the highest-priority ready thread. However, there are optimization strategies built into the operating system to address situations in which the default scheduling methods would cause problems. The following sections describe these strategies.

### Foreground Process Scheduling

The scheduler runs a foreground process at a higher priority, which means it tends to get more time slices than background processes. In addition, the scheduler ensures that those time slices are longer than the ones allocated to background processes. As a result, the foreground process is much more responsive than other processes because it is scheduled more often, and when it is scheduled it continues running longer before being preempted. By default, Windows 2000 Professional defines short, variable quanta for applications and gives a foreground application a priority boost. On the other hand, Windows 2000 Server has longer, fixed quanta with no priority boost for foreground applications, allowing background services to run more efficiently. To see foreground process scheduling in action, monitor the processor time for a process and move its window to the bottom of the stack. Note that the time value allocated to that process falls immediately. Then move the process to the top of the stack and note that the processor time value rises immediately. See Figure 7.18 for an illustration.

### Automatic Priority Boost

The operating system automatically boosts the thread's priority enough for the low-priority thread to complete its operation and release the resource. After raising a thread's dynamic priority, the scheduler reduces that priority by one level each time the thread completes a time slice (quantum), until the thread drops back to its base priority.

For information about how you can change thread priority, see "Determining and Tuning Priority" later in this chapter.

## Determining and Tuning Priority

If a system has high rate of CPU utilization, it is generally best to add processing power by upgrading to a faster processor or to add a processor for symmetric multiprocessing (SMP). However, if you find that some threads are consistently unable to get processor time, you can adjust the thread's priority to allow them to run temporarily. Adjusting thread priority is not recommended as a long-term solution but is a useful illustration of the effect of thread priority on thread activity. Although this section covers elevating priority to allow a process to run, you can also lower the priority of a process if you want it to run in the background while nothing else is running.

Windows 2000 and the *Windows 2000 Resource Kit* companion CD include several utilities for monitoring the base priority of processes and threads and the dynamic priority of threads. They include:

- In Windows 2000: System Monitor and Task Manager.
- With Support Tools on the Windows 2000 operating system CD: Pviewer.

**Warning** Changing priorities might destabilize the system. Increasing the priority of a process might prevent other processes, including system services, from running. Decreasing the priority of a process might prevent it from running, not merely force it to run less frequently. In addition, lowering priority does not necessarily reduce the amount of processor time a thread receives; this happens only if it is no longer the highest-priority thread.

## System Monitor

System Monitor lets you watch and record—but not change—the base and dynamic priorities of threads and processes. System Monitor has priority counters on the Process and Thread objects:

- Process\Priority Base
- Thread\Priority Base
- Thread\Priority Current

Figure 7.18 is a chart of the base priorities of several processes. The Idle process (the white line at the bottom of the chart) runs at a priority of Idle (0), so it never interrupts another process.

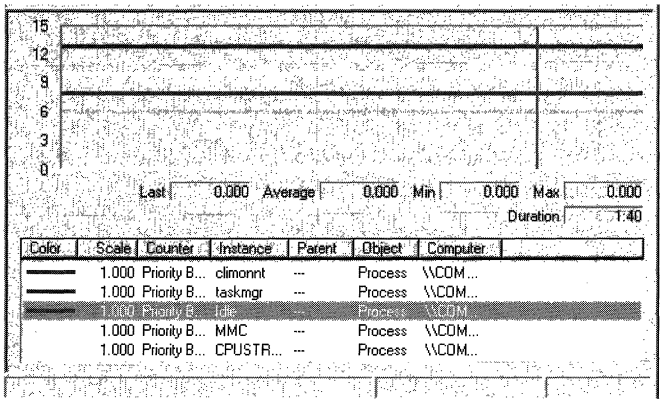



**Figure 7.18 Processes and Their Base Priorities**

Figure 7.19 is a chart of the dynamic priority of the single thread in the Paintbrush utility (Mspaint.exe) as it changes in response to user actions. The base priority of the thread is 8 (Foreground Normal). During this period of foreground use, the dynamic priority of the thread is 14, but drops to 8 when other processes need to run.

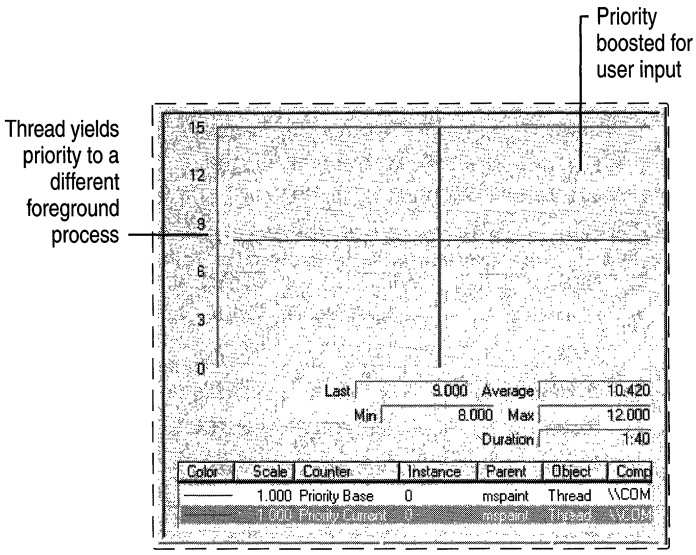



**Figure 7.19 Processes Showing Base and Current Priorities**

## Task Manager

Task Manager presents and lets you change the base priority of a process, but it does not monitor threads. Base priorities changed with Task Manager are effective only as long as the process runs. For more information, see Task Manager online Help or see "Overview of Performance Monitoring" in this book.

---

**Important** In order to make this change, your user account must have permission to increase scheduling priority. By default, only user accounts in the Domain Admins or Administrator groups in Windows 2000 Server or Power User groups in Windows 2000 Professional have this permission.

---

▶ **To change the base priority of a process**

1. In Task Manager, click the **Processes** tab.
2. Right-click a process name. Its menu appears.
3. Click **Set Priority**, and then click a new base priority.

---

**Caution** Problems can arise if multiple processes are running at the High priority class level. Avoid setting more than one process to run at this level.

---

The change in priority is effective at the next Task Manager update; you need not restart the process.

## Process Viewer

Process Viewer (Pviewer.exe), one of the Support Tools on the Windows 2000 operating system CD, lets you monitor process and thread priority and change the base priority class of a process. For information about installing and using the Windows 2000 Support Tools and Support Tools Help, see the file Sreadme.doc in the \Support\Tools folder of the Windows 2000 operating system CD.

## Start Command

When you begin processes from a command prompt using the **start** command, you can specify a base priority for the processes for that run. To see all of the start options, type **start /?** at the command prompt.

You can change the priority class of a process to Real Time by using **start /realtime** at the command prompt or by using Task Manager. Note that changing the priority from **/normal** to **/high** or to **/realtime** can cause severe degradation of performance of other tasks. Note also that setting a processor-bound application to Real Time priority could cause the computer to stop responding.

# Windows 2000 Configuration and Process Priority

Windows 2000 Professional is configured by default to assign variable, short quanta to applications and to boost applications in the foreground. In contrast, Windows 2000 Server is configured to assign long, fixed quanta without any foreground boost to support the efficiency of background services. You can simulate the behavior of one type of operating system when the other type of system is installed—that is, simulate Windows 2000 Server behavior on a Windows 2000 Professional installation, or vice versa. Figure 7.20 shows the interface that you use for changing application response on an installation of Windows 2000 Server.

▶ **To access the Performance Options dialog box**

1. In Control Panel, double-click **System**.
2. Click the **Advanced** tab, and then click **Performance Options**.

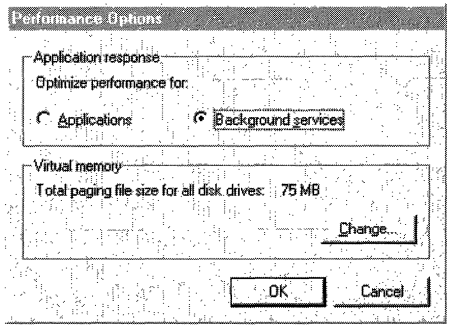



**Figure 7.20 Performance Options Dialog Box in System Properties**

It is easiest to see a change in thread priority as a result of resetting **Performance Options** if you shut down and then restart the computer. Changes in quantum type and length are so minute they are often undetectable to the user.

## Testing Priority Changes

The previous sections described changing thread priority so that, under bottleneck conditions, threads can run more efficiently. Unfortunately, when processor capacity is already stretched to its limit, boosting priorities of blocked threads might not eliminate or reduce processor bottlenecks. In this case, it is best to add processor capacity.

Figure 7.21 shows threads of different priorities contending for processor time. It demonstrates the changing distribution of processor time among processes of different priorities as demand for processor time increases. (This test was conducted using the utility CPU Stress.)

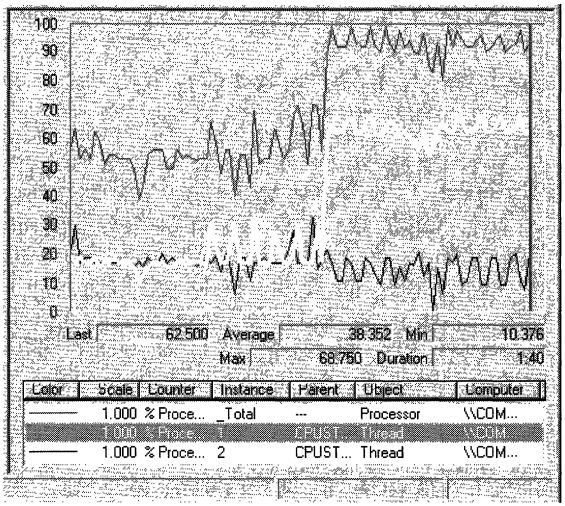



**Figure 7.21 CPU Time Allocation to Threads Based on Priorities**

This chart shows two threads of the same process running on a single-processor computer. Notice the values for processor time for the _Total instance and for threads 1 and 2.

Figure 7.22 shows two threads of the same process running on a single-processor computer. Figures 7.23, 7.24, and 7.25 provide more detail about how processor usage by threads of CPUStres changed in relation to each thread's priority.

In Figure 7.22, the two threads running CPUStres start out at the same low level of activity and run at the same priority—Normal.

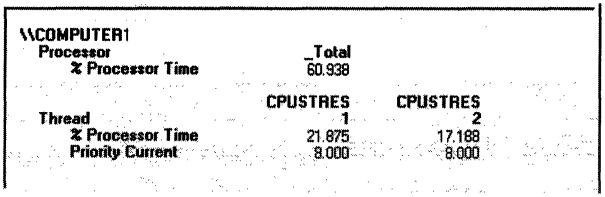



**Figure 7.22 Comparison of Threads at Normal Priority**

Then, if you increase the priority of Thread 1 to Above Normal, and increase its activity level to moderate, you should notice a slight drop in CPU time for Thread 2, as shown in Figure 7.23.

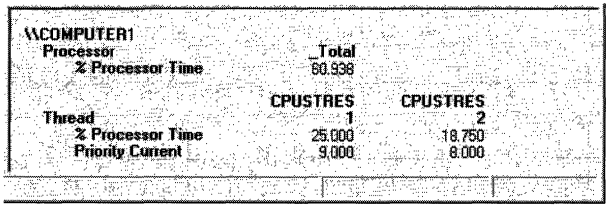



**Figure 7.23 Comparison of Threads at Normal and Above Normal Priority**

Resetting priority for both threads to Normal while running at a higher rate of activity causes each process to consume a large share of processor time, as shown in Figure 7.24. Total processor usage is consistently and extremely high.

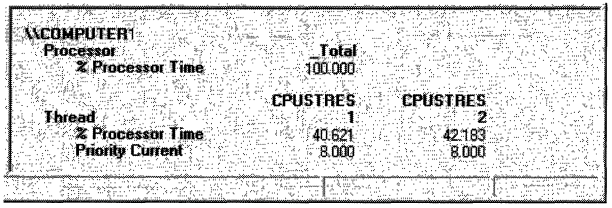



**Figure 7.24 Comparison of Normal-Priority Threads Under Slightly Different Loads**

Finally, raising the priority level of Thread 1 to Above Normal while maintaining its heightened level of activity results in a much greater allocation of processor time to the higher-priority thread and a dramatic drop in processor time for Thread 2, as shown in Figure 7.25.

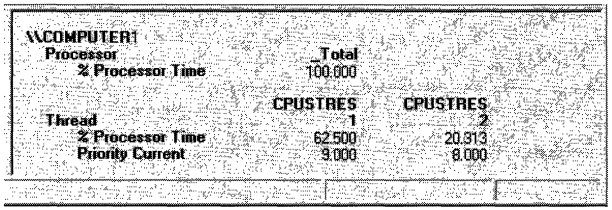



**Figure 7.25 Normal Threads Under Substantially Different Loads**

These results demonstrate that when the processor has extra capacity, increasing the priority of one thread has little effect on the processor time allotted to each of the competing threads. However, when the processor is at its busiest, increasing the priority of one of the threads, even by one priority level, causes the higher-priority thread to get the vast majority of processor time.

In fact, when all processor time is consumed, Thread 2 might not have been scheduled at all were it not for priority boosts. Windows 2000 uses priority boosts to give processor time to lower-priority ready threads that would not otherwise be able to run. This is especially useful when a thread in low priority is waiting for an I/O operation.

# Eliminating a Processor Bottleneck

If you determine that you do have a processor bottleneck, some of the following steps can shorten the processor queue and reduce the burden on your processor. Monitor processor usage and processor queue length after every change to determine the impact on resource usage and overall system operation.

- Upgrade to a faster processor. A faster processor improves response time and throughput for any type of workload.

  Make sure to use a processor with the largest processor cache that is practical. The size of the processor cache is important for your system's performance. You can typically choose from 512 KB to 2 MB for the L2 cache. (The primary cache is determined by what type of processor is installed.)

- Add another processor. If the process you are running has multiple, active threads that are processor-intensive, then it is a prime candidate for a multiprocessor system. It is important that most of the threads be active while the process is running; otherwise, the additional processing power might be wasted. To be certain the process will benefit from an additional processor, verify that most threads are active (that is, consuming a moderate to high amount of processor time). You can see this by monitoring thread state.

  For more information on upgrading to multiple processors, see "Measuring Multiprocessor System Activity" in this book.

- Analyze the application and optimize it if necessary by using the performance utilities in the Platform SDK.
- Upgrade your network or disk adapters. In general, 32-bit, intelligent adapters are recommended. Intelligent adapters provide better overall system performance because they allow interrupts to be processed on the adapter itself, relieving the processor of this work.

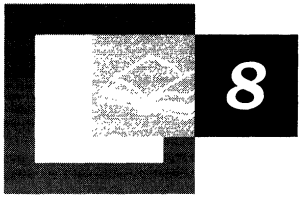
  Try to obtain adapters that have optimization features such as interrupt moderation and features for networking, such as card-based TCP/IP checksum support.

# Additional Resources

- For more information about how Windows 2000 manages processes and threads, including a discussion of its scheduling strategies, see *Inside Windows NT, Second Edition*, available from Microsoft Press.

CHAPTER 8

# Examining and Tuning Disk Performance




The disk system handles the storage and movement of programs and data on your system, giving it a powerful influence on your system's overall responsiveness. The Performance console provides disk-specific counters that enable you to measure disk activity and throughput, and instructs you on strategies to improve disk performance. In addition, it also covers tools found on the *Microsoft® Windows® 2000 Resource Kit* companion CD that can assist you in determining which programs are putting the greatest demand on your disk system.

### In This Chapter

Disk Monitoring Concepts 367
Configuring the Disk and File System for Performance 367
Working with Disk Counters 370
Establishing a Baseline for Disk Usage 381
Investigating Disk Performance Problems 385
Resolving Disk Bottlenecks 389
Evaluating Cache and Disk Usage by Applications 390

### Related Information in the Resource Kit

- For more information about disks and file systems, see "Disk Concepts and Troubleshooting" and "File Systems" in this book.
- For more information about system monitoring, see "Overview of Performance Monitoring" in this book.
- For more information about evaluating memory usage, see "Evaluating Memory and Cache Usage" in this book.

# Disk Monitoring Concepts

You need to observe many factors in determining the performance of a disk system. These include the level of utilization, the rate of throughput, the amount of disk space available, and whether a queue is developing for their disk systems. It is also important to monitor other types of activity that arise from disk operations, such as interrupts generated by the disk system and paging activity, because of their influence on other resources, such as processor or memory.

Figure 8.1 illustrates the importance of monitoring disk systems in relation to the overall performance of your system.

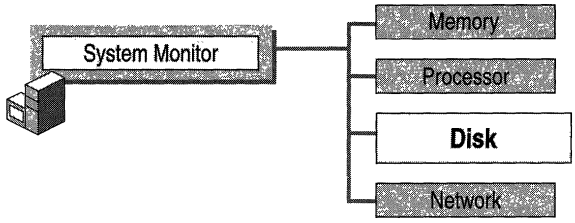



**Figure 8.1 Role of Disk Monitoring in System Monitoring**

Many of these factors are interrelated. For example, if utilization is high, transfer rates (*throughput*) might peak, and a queue might begin to form. These conditions might result in increased response time and cause performance to slow. Although disk space doesn't directly affect the disk's transfer rate, when extremely low, disk space can also have an influence on response time because applications that read and write data can't do so as efficiently. Detect these performance issues through monitoring before they cause problems.

This chapter covers the following stages of a disk-monitoring strategy:

- Configuring the disk and file system for best performance
- Working with disk counters to monitor your disk space and disk efficiency
- Establishing a baseline for disk performance
- Investigating performance problems

# Configuring the Disk and File System for Performance

The type of file system you are using affects your disk performance. The way in which the disk aligns tracks and sectors during format also affects performance. You should configure the file system and the disk's track and sector alignment for optimal performance.

# Configuring Your File System

In general, the NTFS file system is the recommended file system for its advantages in terms of reliability and security, and it is required for large drive sizes. However, these advantages come with some overhead. You can disable some functionality to improve NTFS performance as follows:

- Disable creation of short names. By default, NTFS generates the style of file name that consists of eight characters, followed by a period and a three-character extension for compatibility with MS-DOS and Microsoft Windows 3.*x* clients. If you are not supporting these types of clients, you can turn off this setting by changing the default value of the **\NtfsDisable8dot3NameCreation** registry entry (in HKEY_LOCAL_MACHINE \SYSTEM\CurrentControlSet\Control\Filesystem) to 1.
- Disable last access update. By default NTFS updates the date and time stamp of the last access on directories whenever it traverses the directory. For a large NTFS volume, this update process can slow performance. To disable automatic updating, change the value of the **NtfsDisableLastAccessUpdate** registry entry (in HKEY_LOCAL_MACHINE\SYSTEM\CurrentContolSet\Control \Filesystem)\ to 1.

- Reserve appropriate space for the master file table. Add the **NtfsMftZoneReservation** entry to the registry as a REG_DWORD in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\FileSystem. When you add this entry to the registry, the system reserves space on the volume for the master file table. Reserving space in this manner allows the master file table to grow optimally. If your NTFS volumes generally contain relatively few files that are typically large, set value of this registry entry to 1 (the default). Typically you can use a value of 2 or 3 for moderate numbers of files, and 4 (the maximum) if your volumes tend to contain a relatively large number of files. However, make sure to test any settings greater than 2 because these higher values cause the system to reserve a much larger portion of the disk for the master file table.

**Caution** Do not use a registry editor to edit the registry directly unless you have no alternative. The registry editors bypass the standard safeguards provided by administrative tools. These safeguards prevent you from entering conflicting settings or settings that are likely to degrade performance or damage your system. Editing the registry directly can have serious, unexpected consequences that can prevent the system from starting and require that you reinstall Windows 2000. To configure or customize Windows 2000, use the programs in Control Panel or Microsoft Management Console (MMC) whenever possible.

For information about changing the registry, see Windows 2000 online Help.

## Bypassing I/O Counts

By default, Task Manager continuously measures data for process I/O operations that you can select and display under the Processes tab in Task Manager. In a multiprocessor environment, this data is shared by the processors on which the process runs. When a process that generates considerable disk and network I/O, such as a database service, runs on several processors, updating the shared measurements of process I/O and global I/O operations can slow the system. You can improve the performance of I/O-intensive operations on SMP systems if you configure the system to bypass the global I/O counters and Task Manager process I/O counters. To do so, add the **CountOperations** entry to the registry as a REG_DWORD in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\I/O System\. Set the entry value to 0. When so configured, Task Manager no longer provides per-process I/O measurements. For more information about Task Manager, see "Overview of Performance Monitoring" earlier in this book.

## Configuring Disk Alignment

Windows 2000 has an internal structure called the *master boot record* (MBR) that limits the maximum number of hidden sectors to 63. (For more information about the master boot record, see "Disk Concepts and Troubleshooting" in this book.) This characteristic of the MBR causes the default starting sector for disks that report more than 63 sectors per track to be the 64th sector. As a result, when programs transfer data to or from disks that have more than 63 sectors per track, misalignment can occur at the track level, with allocations beginning at a sector other than the starting sector. This misalignment can defeat system optimizations of I/O operations designed to avoid crossing track boundaries.

Additional disk-design factors make proper alignment even more difficult to achieve. For example, track information reported by disks is not always accurate. In addition, many disks have different numbers of sectors on different tracks (as might be the case with the outer bands versus the inner bands). Diskpar.exe, a sample program on the *Windows 2000 Resource Kit* companion CD, shows how you can use Windows 2000 APIs to obtain and set partition information. By applying the same functions used in this tool, you can avoid performance loss due to disk misalignment on disks with large track sizes and alignment optimizations.

# Working with Disk Counters

Windows 2000 includes counters that monitor the activity of physical disks (including removable media drives) and logical volumes. The PhysicalDisk object provides counters that report physical-disk activity; the LogicalDisk object provides counters that report statistics for logical disks and storage volumes. These counters measure disk throughput, queue length, usage, and other data. The interrelationships between different aspects of disk performance make it useful to monitor them simultaneously. The operating system enables a driver called Diskperf.sys to activate the disk monitoring counters. By default, the operating system activates only the PhysicalDisk performance counters. Users must activate the LogicalDisk counters manually using the **diskperf** command. See the following procedure for activating disk counters with the **diskperf** command.

▶ **To use the diskperf command to enable LogicalDisk object counters**

- At the command prompt, type **diskperf –yv**

The **diskperf** command takes the following syntax:

**diskperf** [**–y**[**d**|**v**] | **–n**[**d**|**v**]] [*computer_name*]

Use **-y** to enable counters and **-n** to disable counters. To specify the type of counters, include **d** for physical disk drives or **v** for logical disk drives or storage volumes. When the operating system starts up, it automatically sets the **diskperf** command with the **-yd** switch to activate physical disk counters. For more information about using the **diskperf** command, type **diskperf -?** at the command prompt.

The PhysicalDisk object counters provide data on activity for each of the physical disks in your system; the LogicalDisk object counters provide data on logical volumes in your system. The System Monitor user interface identifies physical disks by number starting with 0. If you are monitoring logical disks, it identifies these by drive letter. For logical disks consisting of multiple physical disks, the disk instances might appear as 0 C and disk 1 C, where logical drive C: consists of physical drives 0 and 1.

When monitoring logical volumes, remember that they might share a physical disk and your data might reflect contention between them. If you have a *spanned*, *striped*, or *mirrored* volume with disk controllers that support hardware-enabled redundant array of independent disks (RAID) volumes, the counters report physical disk data for all disks in the stripe or mirror as if they are a single disk. For disks with controllers that use software-enabled RAID, the counters report disk data for each physical disk.

Use the counters described in Table 8.1 to measure disk space, disk throughput, and disk utilization.

**Table 8.1 Performance Objects and Counters for Disk Monitoring**

| Counter | Description |
|---|---|
| LogicalDisk\% Free Space | Reports the percentage of unallocated disk space to the total usable space on the logical volume. When calculating the _Total instance, the %Free Space counters recalculate the sum as a percentage for each disk.<br>There is no % Free Space counter for the PhysicalDisk object. |
| LogicalDisk\|PhysicalDisk \Avg. Disk Bytes/Transfer | Measures the size of input/output (I/O) operations. The disk is efficient if it transfers large amounts of data relatively quickly.<br>Watch this counter when measuring maximum throughput.<br>To analyze transfer data further, use Avg. Disk Bytes/Read and Avg. Disk Bytes/Write. |

*(continued)*

**Table 8.1 Performance Objects and Counters for Disk Monitoring** *(continued)*

| Counter | Description |
|---|---|
| LogicalDisk\|PhysicalDisk \Avg. Disk sec/Transfer | Indicates how fast data is being moved (in seconds). Measures the average time of each data transfer, regardless of the number of bytes read or written. Shows the total time of the read or write, from the moment it leaves the Diskperf.sys driver to the moment it is complete.<br>A high value for this counter might mean that the system is retrying requests due to lengthy queuing or, less commonly, disk failures.<br>To analyze transfer data further, use Avg. Disk sec/Read and Avg. Disk sec/Write. |
| LogicalDisk\|PhysicalDisk \Avg. Disk Queue Length | Tracks the number of requests that are queued and waiting for a disk during the sample interval, as well as requests in service. As a result, this might overstate activity.<br>If more than two requests are continuously waiting on a single-disk system, the disk might be a bottleneck. To analyze queue length data further, use Avg. Disk Read Queue Length and Avg. Disk Write Queue Length. |
| LogicalDisk\|PhysicalDisk \Current Disk Queue Length | Indicates the number of disk requests that are currently waiting as well as requests currently being serviced. Subject to wide variations unless the workload has achieved a steady state and you have collected a sufficient number of samples to establish a pattern.<br>An instantaneous value or snapshot of the current queue length, unlike Avg. Disk Queue Length, Avg. Disk Read Queue Length, and Avg. Disk Write Queue Length, that reports averages. |
| LogicalDisk\|PhysicalDisk \Disk Bytes/sec | Indicates the rate at which bytes are transferred and is the primary measure of disk throughput.<br>To analyze transfer data based on reads and writes, use Disk Read Bytes/sec and Disk Write Bytes/sec, respectively. |
| LogicalDisk\|PhysicalDisk \Disk Transfers/sec | Indicates the number of read and writes completed per second, regardless of how much data they involve. Measures disk utilization.<br>If value exceeds 50 (per physical disk in the case of a stripe set), then a bottleneck might be developing.<br>To analyze transfer data based on reads and writes, use Disk Read/sec and Disk Writes/sec, respectively. |
| LogicalDisk \Free Megabytes | Reports the amount of bytes on the disk that are not allocated.<br>There is no Free Megabytes counter for the PhysicalDisk object. |

*(continued)*

**Table 8.1 Performance Objects and Counters for Disk Monitoring *(continued)***

| Counter | Description |
|---|---|
| LogicalDisk\|PhysicalDisk \Split IO/sec | Reports the rate at which the operating system divides I/O requests to the disk into multiple requests. A split I/O request might occur if the program requests data in a size that is too large to fit into a single request or if the disk is fragmented. Factors that influence the size of an I/O request can include application design, the file system, or drivers. A high rate of split I/O might not, in itself, represent a problem. However, on single-disk systems, a high rate for this counter tends to indicate disk fragmentation. |
| LogicalDisk\|PhysicalDisk \% Disk Time | Reports the percentage of time that the selected disk drive is busy servicing read or write requests. Because this counter's data can span more than one sample, and consequently overstate disk utilization, compare this value against % Idle Time for a more accurate picture.<br><br>By default this counter cannot exceed 100 percent; however, you can reset the registry to allow System Monitor to display percentages exceeding 100 percent if appropriate. For information about this adjustment and other aspects of performance data collection and reporting, see "Performance Objects" in "Overview of Performance Monitoring" in this book. |
| LogicalDisk\|PhysicalDisk \% Disk Write Time | Reports the percentage of time that the selected disk drive is busy servicing write requests. |
| LogicalDisk\|PhysicalDisk \% Disk Read Time | Reports the percentage of time that the selected disk drive is busy servicing read requests. |
| LogicalDisk\|PhysicalDisk \% Idle Time | Reports the percentage of time that the disk system was not processing requests and no work was queued. Notice that this counter, when added to % Disk Time, might not equal 100 percent, because % Disk Time can exaggerate disk utilization. |

When working with the disk-time or disk-queue length counters, be aware of the following limitations that might yield unlikely counter values.

- The % Disk Read Time and % Disk Write Time counters can exaggerate disk time. This is because they report busy time based on the duration of the I/O request, which includes time spent in activities other than reading to or writing from the disk. It then sums up all busy time for all requests and divides it by the elapsed time of the sample interval. If multiple requests are in process at a time, the total request time is greater than the time of the sample interval; as a result, reported disk utilization can exceed actual utilization.
- Counter values that report sums can be misleading for multidisk systems. When you look at the _Total instance for the % Disk Time or disk-queue counters on a multidisk system, the counters report values totaled for all disks and do not divide these totals over the number of disks in use. Therefore, in a system with one idle disk and one disk that is 100 percent busy, it can appear as if all disks are 100 percent busy.

The following sections describe how you can use disk-monitoring counters to observe available space on the disk and to observe the efficiency of disk operations as you become acquainted with your system's disk performance.

# Monitoring Disk Space

It is important to monitor the amount of available storage space on your disk because programs might fail due to an inability to allocate space. In addition, low disk space might make it impossible for your paging file to grow to support virtual memory. Fragmentation also has this effect. For information about setting the paging file size for optimal performance, see "Evaluating Memory and Cache Usage" in this book.

Use the % Free Space and Free Megabytes counters to monitor disk space. If the available space is becoming low, you might want to run Disk Cleanup in the Disk Properties dialog box, compress the disk, or move some files to other disks. Notice that disk compression incurs some performance loss.

Another option is Remote Storage, which enables you to create virtual disk storage out of tape or optical drives. When you use this service, infrequently accessed files are moved to tape or to other media storage. Remote Storage volumes are well suited for data that you need to access only at certain intervals, such as quarterly reports. For more information about remote storage options, see "Data Storage and Management" in this book.

If you are using NTFS and you want to restrict the amount of space allocated by individual users, use the Quota tab in Disk Properties. Notice that using quotas results in a small performance loss. If you are not using NTFS, you can set an alert on the % Free Space counter to track dwindling disk space.

---

**Note** Free space on a Distributed file system (Dfs) share can change as you move from one directory to another within a Dfs share namespace. Thus, when accessing network shares, do not assume that the amount of free space you see at the root is available throughout the entire tree.

---

Even if you are not currently short on disk space, you need to be aware of the storage requirements for applications you are running. Complete the following procedure to determine whether your disk has adequate space for your needs.

▶ **To evaluate the adequacy of your system's disk capacity**

1. For best results, start with 1 GB (although the minimum disk size required to install the operating system might be lower).
2. Add the total size of all applications.
3. Add the size of the paging file (this depends on the amount of memory; this size should be at least twice that of system memory).
4. Add the amount of disk space budgeted per user (if a multiuser system), multiplied by the number of users.
5. Multiply by 130 percent to allow room for expansion (this percentage can vary based on your expected growth).

The result is the size of disk you need.

---

**Note** Although not exactly a disk-storage issue, disk fragmentation slows the transfer rate and seek times of your disk system and you should monitor for increasing disk fragmentation. On single-disk systems, you can use the Split IO/sec counter to determine the degree of fragmentation of your disks. Defragment the disk if this counter rate is consistently high and run Disk Defragmenter periodically to keep stored data organized for best performance.

---

Figure 8.2 shows a graph of disk counters including % Free Space. Notice that the % Free Space counter begins to rise approximately half way through the graph. This illustration shows changes that result from deleting files on the disk.

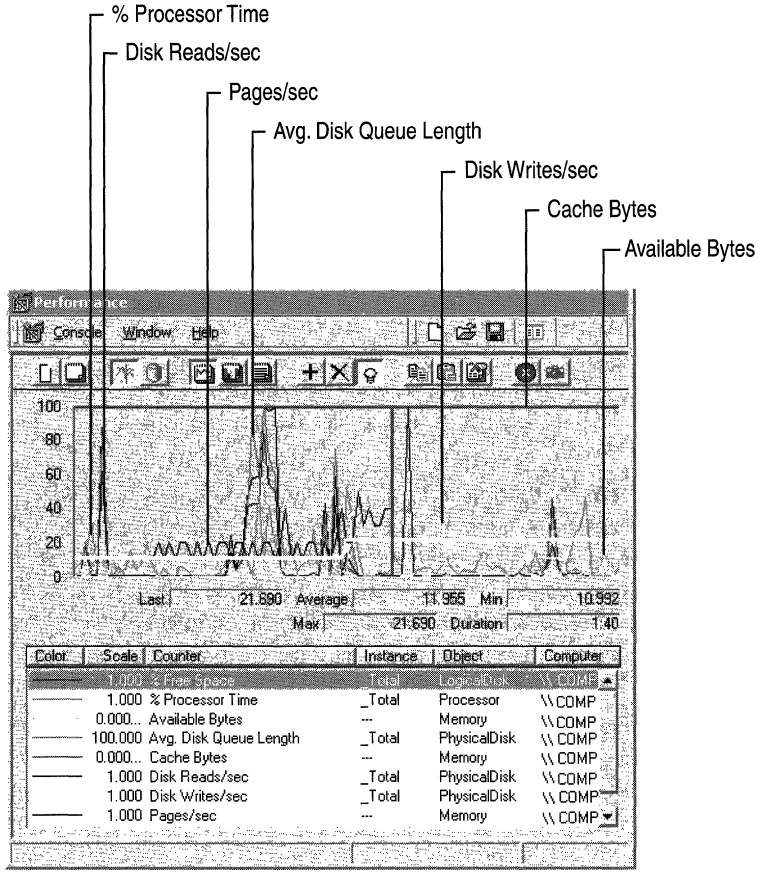



**Figure 8.2 Increase in % Free Space Counter**

# Monitoring Disk Efficiency

Along with disk capacity, you should consider disk throughput when evaluating your starting configuration. Use the bus, controller, cabling, and disk technologies that produce the best throughput that is practical and affordable. Most workstations perform adequately with the most moderately priced disk components. However, if you want to obtain the best performance, you might want to evaluate the latest disk components available.

If your configuration contains different types of disks, controllers, and buses, the differences in their designs can have an influence on throughput rates. You might want to test throughput using these different disk systems to determine if some components produce less favorable results overall or for certain types of activity, and replace those components as needed. In addition, the use of certain kinds of volume-set configurations can offer performance benefits. For example, using stripe sets can provide better performance because they increase throughput by enabling multiple disks to service sequential or clustered I/O requests. System Monitor supports monitoring volume sets with the same performance objects and counters provided for individual disks. Notice that hardware-based RAID devices report all activity to a single physical disk and do not show distribution of disk operations among the individual disks in the array. For more information about using striped volumes, see Windows 2000 online Help.

Be aware of the seek time, rotational speed, access time, and the data transfer rate of your disks by consulting manufacturer documentation. Also consider the bandwidth of cabling and controllers. The slowest component determines the maximum possible throughput, so be sure to monitor each component.

To compare the performance of different disks, monitor the same counters and activity on the disks. If you find differences in performance, you might want to distribute workload to the better performing disk or replace slower performing components.

## Preparing for Comparison Testing

If you want to know more about the volume and rate of activity through the disk system, monitor the reading and writing activity as described in the following sections. Before you begin to test disk efficiency, complete the following steps to ensure valid results:

- When testing disk performance, log performance data to another physical disk or computer so that it does not interfere with the disk you are testing. If you cannot do this, log to another logical volume on the drive, or measure monitoring overhead during an idle period and subtract that overhead from your data to ensure your results include only disk-specific data and not overhead from other activity.
- Monitor individual instances whenever practical. Summed values and the _Total instance can provide overstated values. For more information about interpreting counter data, see "Working with Disk Counters" earlier in this chapter.

- Remember to defragment your disk before testing. If your disk is nearly full, the remaining free space is likely to be fragmented, which adds to the seek time of I/O write operations as the disk looks for each sector of free space.
- Ensure that disks being monitored are not compressed or encrypted, to avoid having these features add overhead during monitoring. However, if you plan to use these features, testing performance with the features deployed can yield results that are more representative of your production environment.

The following are suggested tests to perform to learn about your disk system's performance.

## Testing Maximum Throughput

A maximum throughput test tells you about one of the limits of your system. To conduct a maximum throughput test, you can use one of the request-generation programs that are publicly available on the World Wide Web. Use the following counters on the PhysicalDisk and LogicalDisk objects for this test:

- Avg. Disk Read Queue Length
- Avg. Disk Bytes/Read
- Avg. Disk sec/Read
- Disk Read Bytes/sec
- Disk Reads/sec

Figure 8.3 illustrates a test of maximum throughput. Notice that the values for Disk Read Bytes/sec and Avg. Disk Read Queue Length become extremely high in this graph.

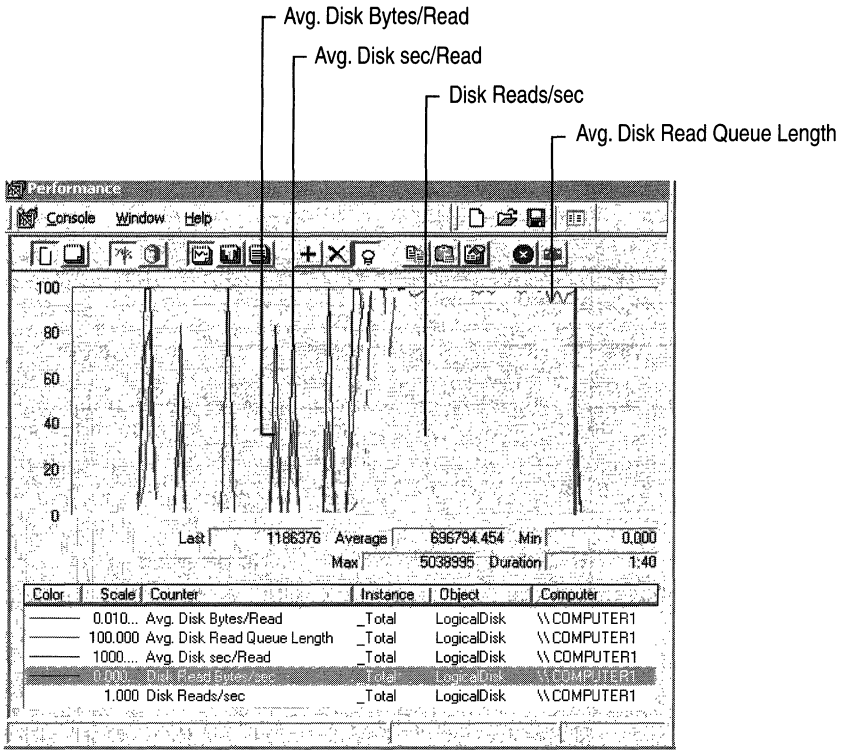



**Figure 8.3 A Disk Reaching Maximum Throughput**

After you determine the maximum throughput for your disk, you can adjust the load on your disk so it does not become a bottleneck.

## Testing Reading vs. Writing

Some disks and disk configurations perform better when reading than when writing. You can compare the reading and writing capabilities of your disks by reading from a physical disk and then writing to the same physical disk. To measure reading from and writing to disk, log the Logical and Physical Disk objects in System Monitor, then chart the counters shown in Table 8.2.

**Table 8.2 Counters for Measuring Reading and Writing**

| For information on | Counters for reads | Counters for writes |
|---|---|---|
| Average size of the request | Avg. Disk Bytes/Read | Avg. Disk Bytes/Write |
| Average duration of the request | Avg. Disk sec/Read | Avg. Disk sec/Write |
| Rate of transfer for the type of request | Disk Read Bytes/sec | Disk Write Bytes/sec |
| Rate at which requests are processed | Disk Reads/sec | Disk Writes/sec |

You might see some variations in the time it takes to read from or write to disk on standard disk configurations. For example, disks with fast write caches can complete write operations very quickly if there is sufficient idle time between random writes. Also, if reads are sequential, read operations might also occur very quickly, provided the disk has had time to prefetch data. Prefetching data is the process whereby data that is expected to be requested is read ahead into the onboard cache.

On RAID 5 volumes and stripe sets, reading is faster than writing. When you read, you read only the data; when you write, you read, modify, and write the *parity*, as well as the data. The exception to this rule is full-stripe writes. If entire stripes are being written, there is no need to read the old data or parity.

Mirrored volumes also are usually faster at reading than writing, and are faster at writing than RAID 5 volumes. When a mirrored subsystem gets a read request, it chooses one of the two or more disk sources to service the request, in a round-robin fashion, or based on disk utilization.

When you start writing to the disk during a read operation that you are monitoring, you will notice some dips in the curves of graphed data for read activity. This is because the application doing the reads must stop briefly to allow the write operation to proceed and then, when the write is finished, the read operation resumes. You can observe this as Performance Logs and Alerts service logs data.

Figure 8.4 shows the effect of writing on the efficiency of the reads. Notice how the increase of reading activity is accompanied by a slight decrease in writing.

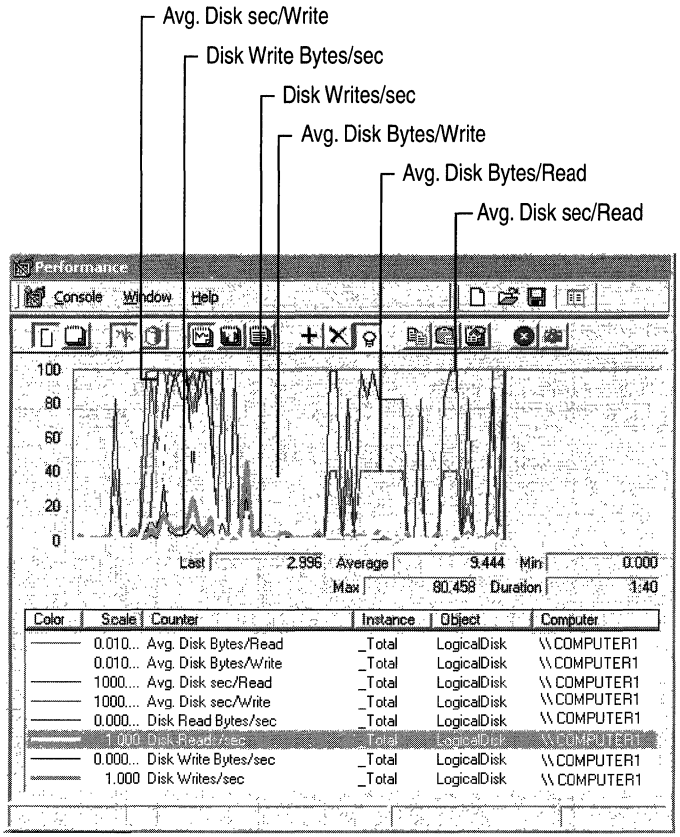



**Figure 8.4 How I/O Operations are Affected by Competing Activity**

# Establishing a Baseline for Disk Usage

Start your disk monitoring process by establishing a *baseline*, which is the level of performance you can expect during typical usage and workloads. Establishing a baseline consists of collecting and analyzing data about typical disk usage under typical disk load.

# Data Collection

Monitor disk counters along with counters from other objects. The following is a list of recommended counters.

- LogicalDisk\% Free Space
- PhysicalDisk\Disk Reads/sec
- PhysicalDisk\Disk Writes/sec
- PhysicalDisk\Avg.Disk Queue Length
- Memory\Available Bytes
- Memory\Cache Bytes
- Memory\Pages/sec
- Processor(*All_Instances*)\% Processor Time
- System\Processor Queue Length

For server computers add networking counters. These include the following:

- Network Segment\% Net Utilization (you must enable the Network Monitor Driver protocol to use this counter).
- Transmission counters such as Segments/sec, Datagrams/sec, or Frames/sec (for the object installed by the networking protocol in use).
- Application counters such as Server\Bytes Transmitted/sec or Server\Bytes Received/sec for the Server service

Figure 8.5 depicts a typical display for collecting overall system performance data.

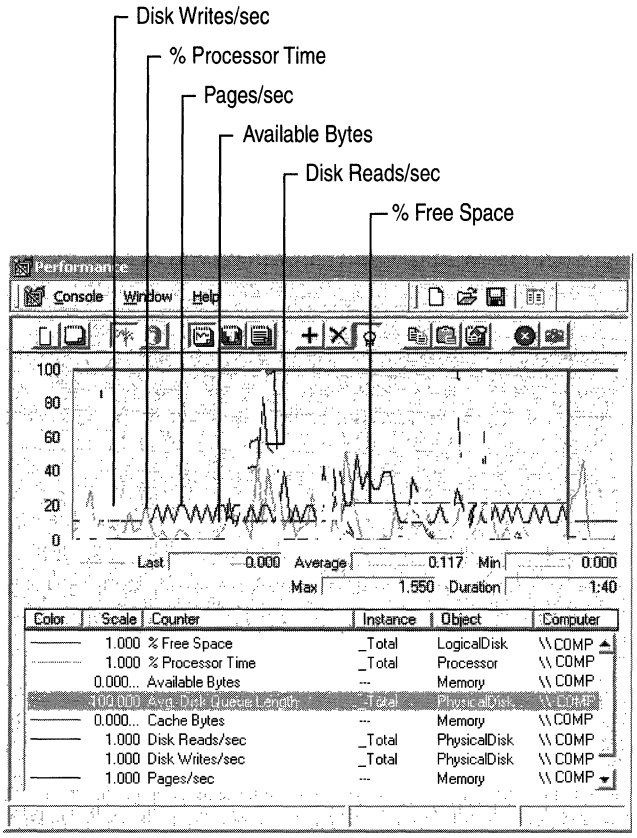



**Figure 8.5 Counter Configuration for Baseline Monitoring**

Observe activity at various times of day over a range of intervals, starting with one day, one week, one month, and so on. Over time a pattern develops, and you can see that the data tends to fall consistently within a particular range of values —that resulting range is your baseline.

You can monitor for short intervals such as two to five seconds, if your workload is characterized by random bursts of heavy activity. Otherwise 60-second intervals are adequate. If system demands fluctuate during the day, you might want to take shorter samples during periods of heaviest activity and longer samples when activity is tapering off.

For best results during monitoring, try to isolate the disk so that workload unrelated to your test does not affect your results. If you are logging performance to a disk that you are monitoring, values for the disk reflect a small amount of writing activity for that logging.

# Data Analysis

While analyzing values at specific times, notice the type of work being performed on your system. Knowing the schedule and nature of your workload is important if you need to reschedule that work or distribute to other systems for better performance.

When interpreting log data, remember the limitations of the performance counters that report sums or that report disk time. The counters sum the totals rather than recalculate them over the number of disks. In addition, disk-time percentage counters cannot exceed 100 percent. Instead, use the Avg. Disk Queue Length, Avg. Disk Read Queue Length, and Avg. Disk Write Queue Length counters to display disk activity as a decimal, rather than a percentage, so that it displays values over 1.0 (100 percent). Then, remember to recalculate the values over the whole disk configuration.

---

**Note** Although disk-time percentage counters cannot exceed 100 percent by default, you can reset the registry to allow System Monitor to display percentages exceeding 100 percent if appropriate. For information about this adjustment and other aspects of performance data collection and reporting, see "Performance Objects" in "Overview of Performance Monitoring" in this book.

---

In general, you can exclude spiking values from your baseline, but make sure you understand what causes them. For example, if you run a weekly backup every Friday night it is acceptable to see out-of-range disk values during that time. But it is important that you know why the spikes are happening. If the pattern starts to shift or you feel that the baseline performance is not satisfactory, use additional counters to monitor disk activity and usage as described in the following sections. You might need to upgrade resources as described in "Resolving Disk Bottlenecks" later in this chapter. If you have access to source code for applications that are in use, you might want to fine-tune these for more efficient data access.

When counter values fall outside the range established for your baseline, follow the instructions contained in "Investigating Disk Performance Problems" later in this chapter. If you encounter a problem or need information about how to improve performance, see "Resolving Disk Bottlenecks" later in this chapter.

# Investigating Disk Performance Problems

Several conditions must exist in order for you to determine that a disk bottleneck exists. These are a sustained rate of disk activity well above your baseline, persistent disk queues longer than two per disk, and the absence of a significant amount of paging. Without this combination of factors, it is unlikely that you have a bottleneck. However, if you suspect a disk-specific performance problem, monitor the following types of counters:

- Paging counters (under the Memory object): Pages/sec, Page Reads/sec, Page Writes/sec
- Usage counters: % Disk Time, % Disk Read Time, % Disk Write Time, % Idle Time, Disk Reads/sec, Disk Writes/sec, Disk Transfers/sec
- Queue-length counters: Avg. Disk Queue Length, Avg. Disk Read Queue Length, Avg. Disk Write Queue Length, Current Disk Queue Length
- Throughput counters: Disk Bytes/sec, Disk Read Bytes/sec, Disk Write Bytes/sec

---

**Note** Although not reflected in disk activity, the rate of interrupt generation by your disk hardware can have a systemwide performance impact. Disk I/O can sometimes generate a sufficient number of interrupts to slow the performance of the processor. Although this does not constitute a "disk" bottleneck, it *is* a processor bottleneck caused by the disk system that can slow the responsiveness of the whole computer. For more information about monitoring disk interrupts and reducing their impact on system performance, see "Analyzing Processor Activity" in this book.

---

The following sections describe how you interpret the values of these counters to reveal or rule out a bottleneck.

## Monitoring Paging

The symptoms of a memory shortage are very similar to those of a disk bottleneck. When physical memory is scarce, the system starts writing the contents of memory to disk and reading in smaller blocks more frequently (this process is called *paging*). The less memory you have, the more the disk is used, resulting in a greater load on the disk system. Therefore, it's important to monitor memory counters along with disk counters when you suspect a performance problem with your disk system.

Monitor paging activity along with disk reading and writing, using the following counters:

- Avg. Disk Queue Length
- Disk Reads/sec
- Disk Writes/sec
- Memory\Pages/sec
- Memory\Page Reads/sec
- Memory\Page Writes/sec

Figure 8.6 shows how a memory shortage can cause disk counters to indicate a problem.

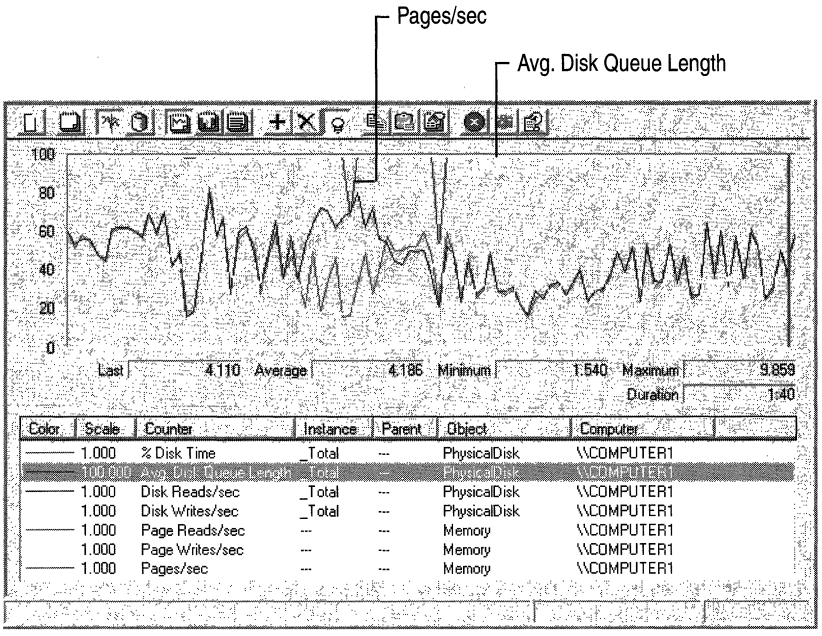



**Figure 8.6 Paging Activity Compared with Disk Activity**

Notice that this figure shows a long disk queue, accompanied by a high rate of paging. Compare the number of page reads against the number of disk reads to see how many times the system accessed the disk to retrieve pages that were not found in memory, or to write pages to free up memory for new data coming in from the disk. When these values are high, the system does not have sufficient memory. Without inclusion of the memory counters to reveal this behavior, you might have assumed that the disk was inadequate. Upgrading the disk in this situation would not have cured the problem.

For more information about measuring memory and identifying memory shortages, see "Evaluating Memory and Cache Usage" in this book.

# Monitoring Usage

In general, a high-performance disk is capable of about 50 to 70 random or up to 160 sequential I/O operations per second. The components you are using, as well as the request size, bus speed, and other factors, determine your system's capacity. Judge the maximum acceptable usage that your system can sustain based on your experience. Disk-time values should not consistently exceed the rate you've established as your baseline for performance. Consistent values in the 70 percent to 85 percent range are a definite cause for concern. However, if a queue is developing, lower percentages might indicate a disk that is unable to handle the load. If you see extremely high rates of disk usage, investigate the factors that might be responsible. Monitoring Disk Transfers/sec (a counter with values equal to the sum of Disk Reads/sec and Disk Writes/sec) or the individual counters Disk Reads/sec and Disk Writes/sec can show you the number of requests for service by the disk; the values of these counters provide a measure of disk demand.

If your workloads consist of random bursts of high activity, you might see high activity rates followed by long periods of idle time. If you only look at the average counter values with these types of workload, it can appear that your disk isn't very busy even though it was bottlenecked during those bursts of high activity. To determine how well your disk system is handling these bursts, sample at short intervals when the activity occurs.

---

**Note** The disk time counters can yield inaccurate values when multiple disks are in use. You can compensate for this by monitoring % Idle Time and comparing its value with the values reported by the % Disk Time, % Disk Read Time, and % Disk Write Time counters.

---

Figure 8.7 depicts maximum disk usage and the development of a queue.

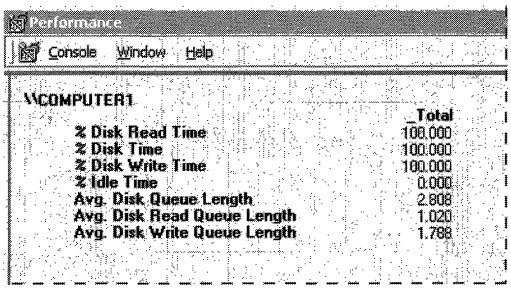



**Figure 8.7 High Disk Time Values**

# Monitoring Queue Length

To determine the number of I/O requests queued for service, track Avg. Disk Queue Length for LogicalDisk or PhysicalDisk. Notice that this might overstate the true length of the queue, because the counter includes both queued and in-service requests. If the value of Avg. Disk Queue Length exceeds twice the number of spindles, then you are likely developing a bottleneck. With a volume set, a queue that is never shorter than the number of active physical disks indicates that you are developing a bottleneck.

Figure 8.8 shows a disk bottleneck with high disk usage and a long queue.

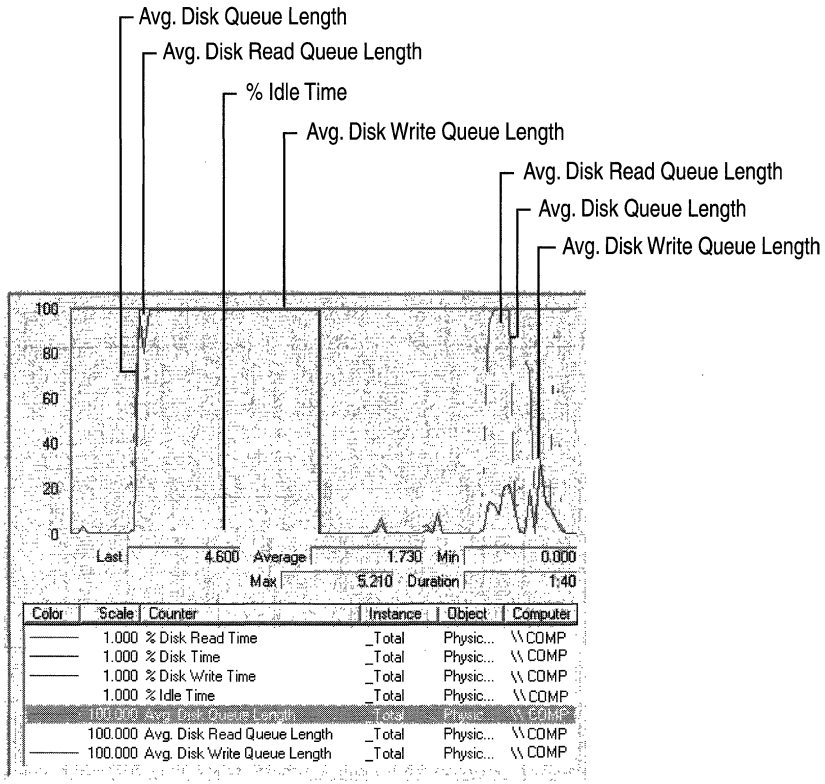



**Figure 8.8 High Disk Usage and a Long Queue**

# Resolving Disk Bottlenecks

A disk that is developing a bottleneck might cause the entire system to slow. If you determine that disk resources are responsible for an overall decrease in system performance, you need to find a solution quickly. Although you might assume that installing another disk offers a quick fix, the right solution to your performance problem depends on the cause of the bottleneck.

Adding a disk is an appropriate solution if you can move files to a new disk, if you can create a stripe set, or if you are out of space. However, for disk-space problems only, you might want to compress your drive, provided that the processor has available cycle to handle the compression activity and that your disk requests are typically large. For a multiuser system, you can also implement disk quotas to restrict growth of user files.

If your disk system is too slow, consider the following alternative steps:

- Rule out a memory shortage. When memory is scarce, the Virtual Memory Manager writes more pages to disk, resulting in increased disk activity. Because low memory is a common cause of bottlenecks, make sure that this is not the source of the problem before adding hardware. Also, make sure to set the paging file to an appropriate size as described in "Evaluating Memory and Cache Usage" in this book.
- Defragment the disk using Disk Defragmenter. For information about using Disk Defragmenter, see Windows 2000 online Help.
- Use Diskpar.exe on the *Windows 2000 Resource Kit* companion CD to reduce performance loss due to misaligned disk tracks and sectors.
- Use stripe sets to process I/O requests concurrently over multiple disks. The type you use depends on your data-integrity requirements. If your applications are read-intensive and require fault tolerance, consider a RAID 5 volume. Use mirrored volumes for fault tolerance and good I/O performance overall. If you do not require fault tolerance, implement stripe sets for fast reading and writing and improved storage capacity. When stripe sets are used, disk utilization per disk should fall due to distribution of work across the volumes and overall throughput should increase.

  When using stripe sets, make sure to use the optimal size of stripe for your workload. In general, the stripe size should be a multiple of the size of the average request.

  If you find that there is no increased throughput when scaling to additional disks in a stripe set, you might be experiencing a bottleneck due to contention between disks for the disk adapter. You might need to add an adapter to better distribute the load.

- Place multiple drives on separate I/O buses, particularly if a disk has an I/O-intensive workload.
- Distribute workload among multiple drives. For example, for database applications, you might want to put transaction logs on a separate spindle from data. Notice that writing to a log file is a sequential operation and tends to be more efficient than the random operations typical of accessing data in a database.

  If you are unsure of a suitable distribution for network applications, track users and the files they work with in order to plan an efficient distribution by means of the auditing capability of the NTFS file system. This tells you which disks are getting the most usage and help you determine whether you should redistribute workloads. For more information about auditing, see Windows 2000 Server Help.

  Windows Clustering and Distributed File System provide solutions for load balancing on different drives. For more information about these technologies, see "Distributed File System" in the *Microsoft® Windows® 2000 Server Resource Kit Distributed Systems Guide*.
- Limit your use of file compression or encryption. These features can add some overhead and you should use them sparingly if not required by your enterprise and if performance is critical.
- When obtaining disk systems, use the most intelligent and efficient components available for your disk system, including controller, I/O bus, cabling, and the disk. Upgrade to faster-speed or wider-bandwidth components as necessary. These measures generally decrease transfer time and improve throughput. Use intelligent drivers that support interrupt moderation or interrupt avoidance to alleviate the interrupt activity for the processor due to disk I/O.

# Evaluating Cache and Disk Usage by Applications

If you are an application developer, you might want to know if your programs read and write data efficiently to and from the disk, as well as how they utilize locality and manage the file-system cache. This section provides information to help you identify situations in which you can improve the I/O performance of applications.

## Random and Sequential Data Access

Comparing random versus sequential operations is one way of assessing application efficiency in terms of disk use. Accessing data sequentially is much faster than accessing it randomly because of the way in which the disk hardware works. The seek operation, which occurs when the disk head positions itself at the right disk cylinder to access data requested, takes more time than any other part of the I/O process. Because reading randomly involves a higher number of seek operations than does sequential reading, random reads deliver a lower rate of throughput. The same is true for random writing. You might find it useful to examine your workload to determine whether it accesses data randomly or sequentially. If you find disk access is predominantly random, you might want to pay particular attention to the activities being done and monitor for the emergence of a bottleneck.

For workloads of either random or sequential I/O, use drives with faster rotational speeds. For workloads that are predominantly random I/O, use a drive with faster seek time.

For workloads that have high I/O rates, consider using stripe sets because they add physical disks, increasing the system's ability to handle concurrent disk requests. Notice, however, that stripe sets enabled in software can cause an increase in consumption of the processor. Hardware-enabled RAID sets eliminate this impact on the processor but increase the consumption of processing cycles on the hardware RAID adapter.

---

**Note** Even when an application reads records sequentially, if the file is fragmented throughout the disk or disks, the I/O will not be sequential. If the disk-transfer rate on a sequential or mostly sequential read operation deteriorates over time, run Disk Defragmenter on the disk and test again. When fragmentation occurs, data is not organized in contiguous clusters on the disk. Fragmentation slows performance because back-and-forth head movement is slow.

---

## I/O Request Size

The size of requests and the rate at which they are sent are important for evaluating the way applications work with the disk. If you are an application developer, you can use the counters, such as Avg. Disk Bytes/Read, that reveal these types of information about I/O requests.

It is typically faster and more efficient to read a few large records than many small ones. However, transfer rates eventually peak due to the fact that the disk is moving blocks of data so large that each transfer occurs more slowly—although its total throughput is quite high. Unfortunately, it is not always easy to control this factor. However, if your system is used to transfer many small units of data, this inefficiency might help to explain, though not resolve, high disk use.

Requests should be at least 8 KB, and, if possible, 64 KB. Sequential I/O requests of 2 KB consume a substantial amount of processor time, which affects overall system performance. However, if you can be sure that only 2 KB of data is necessary, doing a 2 KB I/O is the most efficient, because a larger I/O wastes *direct memory access (DMA)* controller bandwidth. As the record size increases, the throughput increases and the transfer rate falls because it takes fewer reads to move the same amount data.

Using 64 KB requests results in faster throughput with very little processor time. Maximum throughput typically occurs at 64 KB, although some devices might have a higher maximum throughput size. When transferring data blocks greater than 64 KB, the I/O subsystem breaks the transfers into 64-KB blocks. Above 64 KB, the transfer rate drops sharply, and throughput levels off. Processor use and interrupts also appear to level off at 64 KB.

## Investigating Disk Usage by Applications

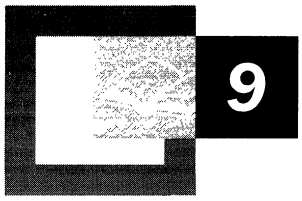
Applications rarely read or write directly to disk. Instead, application code and data is typically mapped into the file system cache and copied from there into the working set of the application. When the application creates or changes data, the data is mapped into the cache and is then written back to the disk in batches. The disk is used only when an application requests a single write-through to disk or it instructs the file system not to use the cache at all for a file, usually because it is doing its own buffering. For this reason, tracking the cache and memory counters provides a way of investigating disk usage by your application. You can find information about monitoring cache and memory counters in "Evaluating Memory and Cache Usage" earlier in this book.

When monitoring disk usage by applications, you might find that applications that submit all I/O requests simultaneously tend to produce exaggerated values for the % Disk Time, % Disk Read Time, % Disk Write Time, and Avg. Disk sec/Transfer counters. Although throughput might be the same for applications that submit I/O requests intermittently, the values of counters that time requests will be much lower. It is important to understand your applications and factor their I/O methods into your analysis.

If you are writing your own tools to test disk performance, you might want to include the FILE_FLAG_NO_BUFFERING parameter in the open call for your test files. This instructs the Virtual Memory Manager to bypass the cache and go directly to disk.

CHAPTER 9

# Monitoring Network Performance




Communication across a network is increasingly important in the work environment. Similar to processors or disks on your system, the behavior of the network has an impact on the operation of your computer. Optimize your system's performance by analyzing network performance, such as monitoring network traffic and resource utilization that affects both hardware and software.

### In This Chapter

Introduction to Network Performance Analysis 395
Tools for Monitoring Network Performance 396
Resolving Network Bottlenecks 425

### Related Information in the Resource Kit

- For more information about the monitoring process, see "Overview of Performance Monitoring" in this book.
- For more information about monitoring memory and cache usage, see "Evaluating Memory and Cache Usage" in this book.
- For more information about monitoring processor performance, see "Analyzing Processor Activity" in this book.
- For more information about monitoring disk system activity, see "Examining and Tuning Disk Performance" in this book.
- For more information about diagnosing system problems, see "Troubleshooting Strategies" in this book.
- For information about the Transmission Control Protocol/Internet Protocol (TCP/IP), see "Introduction to TCP/IP" in the *Microsoft® Windows® 2000 Server Resource Kit TCP/IP Core Networking Guide.*
- For information about the Browser service, see "Windows 2000 Browser Service"in the *Microsoft® Windows® 2000 Server Resource Kit TCP/IP Core Networking Guide.*
- For information about the NetBEUI protocol, see "NetBEUI" in the *Microsoft® Windows® 2000 Server Resource Kit Internetworking Guide.*

# Introduction to Network Performance Analysis

Network performance analysis is a follow-up to other monitoring and tuning efforts that are specific to a workstation or server computer. After you have tested and optimized the client or server system's resources, look at the performance of the network. For information about monitoring and tuning your system's memory, processor, and disk systems, see the optimization chapters of the *Microsoft® Windows® 2000 Server Resource Kit Server Operations Guide* before analyzing the network components.

Figure 9.1 illustrates the sequence for monitoring system performance.

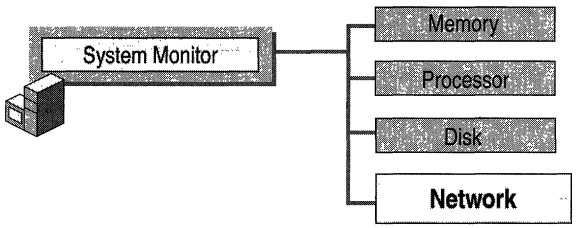



**Figure 9.1 Sequence of Monitoring System Performance**

When you are ready to examine your network's components, start by checking your networking hardware, including elements such as hubs, cables, routers, switches, and network adapters. For information about proper operation, see the manufacturer's documentation.

Use the most current network adapters and device drivers for your network components. In general, you want the widest bandwidth and highest-performing components possible for your entire system and budget. For example, to maximize the benefits of Windows 2000 networking performance enhancements, use adapters that support task offloading capabilities (checksum offloading, IP Security (IPSEC) offloading, or large send offloading) and interrupt moderation.

---

**Note** If you are using media-sense network adapters, Windows 2000 displays an icon in the Taskbar if the adapter becomes disconnected from the network medium. Because the driver supporting the adapter continues to run even when it is not processing traffic, the driver causes the system to continue to use resources unnecessarily. Therefore, you should attend to disconnected adapters when the system reports them. As soon as you see the icon, check the adapter connection. Reconnect the adapter if appropriate; otherwise disable or remove the adapter to avoid the waste of resources associated with this condition.

---

After you have checked the adapters and drivers, verify that your components are properly configured. Set adapters for speeds appropriate to the devices they are using. Notice that autodetection might not select the optimal speed for operation. For proper configuration, see the product documentation.

# Tools for Monitoring Network Performance

Windows 2000 provides two primary utilities for monitoring network performance: System Monitor and Network Monitor. System Monitor, installed with both Windows 2000 Professional and Windows 2000 Server, tracks resource utilization and network throughput. Network Monitor, which you can install on Windows 2000 Server, tracks network throughput in terms of captured network traffic. Network Monitor monitors only local traffic. For monitoring traffic sent to or from any computer on the network, or for remotely capturing frames (for example, over a dial-up network connection) from other computers on the network, upgrade to the version of Network Monitor that ships with Microsoft® Systems Management Server version 2.0, Service Pack 1 or later. To monitor these types of traffic, the system from which you capture must have the Network Monitor driver installed (computers running Windows NT 4.0 must have the Network Monitor Agent version 2.0 installed). There is no support for the Network Monitor driver on Windows 95 or Windows 98.

Use the following tools to examine network traffic and system resource utilization. For information about using System Monitor, see "Overview of Performance Monitoring" in this book.

By default, only an administrator has sufficient privileges to install the Network Monitor driver on a computer to be monitored, to install Network Monitor on the computer used for monitoring, or to start Network Monitor.

## System Monitor

The built-in Performance console provides the ability to monitor network activity along with the other performance data on the system. Treat network components as another set of hardware resources to observe as part of your normal performance-monitoring routine.

Network activity can influence the performance not only of your network components but also of your system as a whole. You should monitor other resources along with network activity, such as disk, memory, and processor activity. System Monitor enables you to track network and system activity using a single tool. Use the following counters as part of your normal monitoring configuration:

- Cache\Data Map Hits %
- Cache\Fast Reads/sec

- Cache\Lazy Write Pages/sec
- Logical Disk\% Disk Space
- Memory\Available Bytes
- Memory\Nonpaged Pool Allocs
- Memory\Nonpaged Pool Bytes
- Memory\Paged Pool Allocs
- Memory\Paged Pool Bytes
- Processor(_Total)\% Processor Time
- System\Context Switches/sec
- System\Processor Queue Length
- Processor(_Total)\Interrupts/sec

Monitoring network activity with System Monitor involves examining performance data at each network layer, as defined in the Open Systems Interconnect (OSI) model; for information about this model, see Appendix A, "OSI Model," in the *Windows 2000 Server Resource Kit TCP/IP Core Networking Guide*. System Monitor provides performance objects for collection of data that reflect transmission rates, packet queue lengths, and other network performance data.

**Note** Because of the overhead of the protocol headers, actual transmission rates might differ from the rates specified for the wire or line in use.

Table 9.1 illustrates the network layers and their associated performance objects.

**Table 9.1 Network Layers and Related Performance Objects**

| OSI layer | Performance objects |
|---|---|
| Application, Presentation | Browser, Server, Redirector, and Server Work Queues |
| Session | NBT Connection (NBT is an abbreviation for NetBT, which means NetBIOS over TCP/IP; NetBIOS stands for network basic input/output system) |
| Transport | Protocol objects: TCP for the Transmission Control Protocol; UDP for the User Datagram Protocol, NetBEUI for NetBIOS, AppleTalk (installed by protocol) |

*(continued)*

**Table 9.1 Network Layers and Related Performance Objects *(continued)***

| OSI layer | Performance objects |
|---|---|
| Network | Network Segment (installed when you install the Network Monitor driver), IP for the Internet Protocol, NWLink IPX/SPX for the Microsoft implementation of Internetwork Packet Exchange/Sequenced Packet Exchange (IPX/SPX). NWLink performance objects display only zeros for counters that report on frame activity.<br><br>On systems running Windows NT version 4.0, installing the Network Monitor Agent installed the Network Segment counters. |
| Data Link, Physical | Network Interface<br><br>These counters are maintained by the driver and can report inaccurate or zero values because of problems with implementation of counters by the driver. |

Begin with the lowest-level components and work your way up as you monitor performance data for your network. Monitor the objects described in this chapter over periods of time ranging from days to weeks, to a month. Using this data, determine a performance baseline, the typical level of performance you expect under typical workloads and usage. A performance baseline gives you a point from which to compare performance over time to identify growth trends, changing demands, or the emergence of a bottleneck. If performance within the baseline range becomes unsatisfactory, tune the network as described in "Resolving Network Bottlenecks" later in this chapter.

As with other resources, establish a baseline for network performance. When performance data is incompatible with your baseline values, investigate the cause. Abnormal network counter values on a server often indicate problems with its memory, processor, or disks. For that reason, the best approach to monitoring a server is to watch network counters in conjunction with Processor\% Processor Time, PhysicalDisk\% Disk Time, and Memory\Pages/sec.

For example, if a dramatic increase in Pages/sec is accompanied by a decrease in Bytes Total/sec handled by a server, the computer is probably running short of physical memory for network operations. Most network resources, including network adapters and protocol software, use nonpaged memory. If a computer is paging excessively, it could be because most of its physical memory has been allocated to network activities, leaving a small amount of memory for processes that use paged memory. To verify this situation, check the computer's system event log for entries indicating that it has run out of paged or nonpaged memory. Also monitor the nonpaged pool memory and overall memory counters. For more information about monitoring memory and performance, see "Evaluating Memory and Cache Usage" in this book.

# Network Counters

Starting from the physical layer and working up to the application layer of the OSI model, you will monitor the performance objects and their counters described in the following sections.

## Network Interface Object

Use the Network Interface object to monitor transmissions starting at the physical layer. The Network Interface object is installed by Transmission Control Protocol/Internet Protocol (TCP/IP) and monitors activity of the IP protocol. The object reports transmissions over the network adapter. There are no separate objects to monitor the adapters over other networking protocols.

When you use the Network Interface object counters, note that the instances include the loopback address, the network adapter, the dial-out wide area network (WAN) wrapper for each device bound under the Routing and Remote Access service, and the dial-up WAN wrapper for each device. The wrapper is code that surrounds network driver interface specification (NDIS) device drivers, providing a uniform interface between protocol drivers and NDIS device drivers and support routines that make the development of an NDIS driver easier.

The instances typically list the loopback address 127.0.0.1 first, and the remaining instances should match the binding order of the TCP/IP protocol. (If Routing and Remote Access does not use IP for a device, its traffic is not counted.) To view the binding order for TCP/IP, in the **Network and Dial-up Connections** dialog box, on the **Connections** menu, click **Advanced**.

Monitor the following Network Interface object counters:

### Network Interface\Output Queue Length

Use this counter to indicate the length of the output packet queue. The value should be low. Queues that are one or two items long constitute satisfactory performance; longer queues mean that the adapter is waiting for the network and cannot keep pace with server requests.

### Network Interface\Packets Outbound Discarded

Use this counter to determine if the network is saturated. If this counter continuously increases, it might indicate that a network is so busy that the network buffers cannot keep up with the outbound flow of packets.

### Network Interface\Bytes Total/sec

Use this counter to determine how the network adapter is performing. The Bytes Total/sec counter should report high values, to indicate a large number of successful transmissions. Compare this value with the value reported by the Network Interface\Current Bandwidth counter, reflecting each adapter's bandwidth. If you see the Bytes Total/sec rate approaching the maximum transfer rate, the probability of collisions on the network increases. This in turn impacts performance by increasing the latency of packet transfer on the network. In this case, you might want to consider increasing the bandwidth or segmenting the network. For example, if using 100 megabit/sec fast Ethernet, and the total rate of bytes transferred per second approaches 65 percent of the maximum network bandwidth, you can improve performance by using a gigabit or faster Ethernet switch to segment the network into smaller networks.

## Network Segment Objects

Use this object to report statistics for the local network segment. To use this object, you must have already installed the Network Monitor driver on the computer where you will run System Monitor and on the computer from which you will collect data. For more information about installing Network Monitor, see "Installing Network Monitor" later in this chapter.

Monitor the following Network Segment object counters:

### Network Segment\Broadcast Frames Received/sec

Use this counter to establish a baseline when monitored over time. To determine the cause of a problem, investigate large variations from the baseline. Because each computer processes every broadcast, frequent broadcasts mean lower performance. Determine what level of broadcasts is excessive based on past performance and your expectations for the local site.

### Network Segment\% Network Utilization

Use this counter to reflect the percentage of network bandwidth used for the local network segment. Use it to monitor the effect of different network operations (such as logon validation or account synchronization). A low value is preferred. This counter should not report values that exceed the maximum recommended for the type of configuration. For example, 30 percent utilization is the maximum recommended for an unswitched Ethernet network. This means that a 10-megabyte (MB) Ethernet network becomes bottlenecked when its throughput exceeds 3 MB per second. If the value of the counter is above 40 percent, collisions can cause problems. You need to determine the appropriate maximum value for this counter based on your network design and topology, and ensure that % Network Utilization does not exceed this limit.

### Network Segment\Total Frames Received/sec

Use this counter to indicate when bridges and routers might be saturated. If network traffic exceeds recommended local area network (LAN) capacity, performance typically suffers across the network. To prevent this situation, it is important to monitor network-wide traffic levels, particularly on larger networks with bridges and routers.

## Network Protocol Objects

When monitoring protocol counters, you are likely to be most concerned with transmission rates. Monitor these rates using counters such as Bytes Total/sec, Datagrams Received/sec and Datagrams Sent/sec, or Frames Received/sec and Frames Sent/sec. When looking at transfer counters, consider the capacity of your network. The value of Bytes Total/sec should not be close to or matching the network capacity, or the network might already be saturated.

Following is a list of typical protocol objects. Monitor the ones that pertain to the network protocol in use.

- For the TCP/IP protocol, use the TCP, IP, UDP, and Internet Control Message Protocol (ICMP) objects. To monitor traffic at the network layer, use the IP object counters Datagrams Forwarded/sec, Datagrams Received/sec, Datagrams/sec, and Datagrams Sent/sec. To monitor activity at the transport layer, use the TCP object counters Segments Received/sec, Segments Retransmitted/sec, Segments/sec, and Segments Sent/sec. ICMP is used for maintaining route tables and diagnosing problems. UDP is used for DNS host name and IP address resolution and for NetBIOS name resolution by a WINS server.

  You should see high values for segments sent and received over the network. If not, reduce broadcast traffic. UDP\Datagrams/sec indicates the number of broadcasts sent and received. This value should be low. If the retransmission rate is high, there might be a hardware problem, so you need to investigate further with system counters such as the processor counters. Use baseline values to determine when these counters are out of range and might indicate a problem.
- For the NWLink protocol, use the following three objects to monitor:
- NWLink IPX for computers communicating over the IPX protocol.
- NWLink NetBIOS for computers communicating over the IPX protocol.

- NWLink SPX for computers connecting over the SPX protocol.

  Each of these objects has the following counters that provide information about network transmissions:

  - Bytes Total/sec
  - Frame Bytes Sent/sec
  - Frame Bytes Received/sec
  - Frames Rejected/sec

  Bytes Total/sec should be high on an active network. Frames Rejected/sec should be low.

  **Note** NWLink performance objects display only zeros for counters that report on frame activity.

- For the NetBEUI protocol, use the NetBEUI and NetBEUI resource objects. Monitor Bytes Total/sec and other transmission counters such as NetBEUI\Frame Bytes Received/sec and NetBEUI\Frame Bytes Sent/sec. In addition, track Frames Rejected/sec for increasing values. Also include NetBEUI Resource\Times Exhausted, which can indicate if resource buffers are being consumed. Information about the resource objects is also recorded in the event log.
- For the AppleTalk protocol, use the AppleTalk object counters.

For information about these counters, see the Performance Counter Reference on the *Windows 2000 Resource Kit* companion CD.

### Improving performance over a slow WAN link under Windows 2000 Server

In general, Windows 2000 is self-tuning, and registry entries related to TCP/IP require no adjustment. If you are using a slow WAN link, adjusting registry entries for TCP/IP can improve performance; however, these changes can adversely affect computers that are short of memory. The following procedure describes how to edit the entries in the registry.

▶ **To edit TCP/IP entries in the registry**

1. On the **Start** menu, click **Run**.
2. In the **Run** dialog box, type **Regedt32**, and then click **OK**.

The following is a list of entries in HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Services\Tcpip\Parameters that have an effect on performance when connecting by means of a slow WAN link. For information about these and other related registry settings, see "Technical Reference to the Windows 2000 Registry" on the *Windows 2000 Resource Kit* companion CD.

**Caution** Do not use a registry editor to edit the registry directly unless you have no alternative. The registry editors bypass the standard safeguards provided by administrative tools. These safeguards prevent you from entering conflicting settings or settings that are likely to degrade performance or damage your system. Editing the registry directly can have serious, unexpected consequences that can prevent the system from starting and require that you reinstall Windows 2000. To configure or customize Windows 2000, use the programs in Control Panel or Microsoft Management Console (MMC) whenever possible.

- **TcpMaxConnectRetransmissions**. The value of this entry can be increased to allow a connection over a slow WAN but should not be set so long that connection attempts never time out. The default is 2 and can range from 0 to 0xFF.
- **TcpMaxDataRetransmissions**. The value of this entry can be lengthened although this time-out is automatically doubled every time a transmission is re-attempted. The default is 5 and can range from 0 to 0xFFFFFFFF.
- **TcpWindowSize**. When modifying this entry, set it to the product of the bandwidth multiplied by the length of the "round trip" between the local computer and the server.
- **MaxUserPort**. The value of this entry can be changed to achieve higher throughput by allowing the creation of more sockets. The value can be set to 0xFFFE. The default is 0x1388. The minimum value is 0x400; values 0 to 0x3FF are reserved for services.
- **MaxHashTableSize**. The value of this entry can be changed to achieve higher throughput by allowing for faster look-up on connections. The default is 0x200 and can range from 0x40 to 0x10000. On a server in a multiprocessor environment, do not increase **MaxHashTableSize** beyond the estimated maximum number of concurrent connections.
- **NumTcbTablePartitions**. The value of this entry can be changed to partition the TCP control block (TCB) table to avoid contention. The default is 0x4; the value should be a power of two, that is, 2, 4, 8, 16, 32, and so on. On multiprocessor systems, change the number of partitions to four times the number of processors in your system.

## NBT Connection Object

Use this object to track session-layer transmissions between computers. NBT stands for NetBT, an abbreviation for NetBIOS over TCP/IP. This feature provides the NetBIOS programming interface over the TCP/IP protocol. It is used for monitoring routed servers that use NetBIOS name resolution.

## Application-Layer Objects

Finally, monitor services or applications at the presentation or application layers. By default, Setup installs the Browser, Redirector, Server, and Server Work Queues objects on computers running Windows 2000. These objects describe performance of file and print services using the Server Message Block (SMB) Protocol.

---

**Note** For detailed information about performance objects and counters, see the Performance Counter Reference on the *Windows 2000 Resource Kit* companion CD.

---

### Browser Object

The primary function of the Browser service is to provide a list of computers sharing resources in a domain along with a list of other domain and workgroup names across the wide area network (WAN). This list is provided to clients that view network resources with My Network Places or the **net view** command. Active Directory replaces the computer browser service used in earlier versions of Windows to provide the Network Basic Input/Output System (NetBIOS) name resolution. The browser service in Windows 2000 provides backward compatibility with client computers that are running earlier versions of Windows.

The Browser performance object consists of counters that measure the rates of announcements, enumerations, and other browser transmissions. If your organization is maintaining domains under Windows NT Server version 4.0, use the counters in Table 9.2 for monitoring the Browser service.

**Table 9.2 Browser Object Counters**

| Counter | Description |
|---|---|
| Browser\Mailslot Allocations Failed | The number of times the datagram receiver has failed to allocate a buffer to hold a user mailslot write. |
| Browser\Mailslot Opens Failed/sec | Indicates the rate of mailslot messages received by this workstation that were to be delivered to mailslots that are not present on this workstation. |
| Browser\Mailslot Receives Failed | Indicates the number of mailslot messages that could not be received due to transport failures. |

*(continued)*

**Table 9.2 Browser Object Counters *(continued)***

| Counter | Description |
|---|---|
| Browser\Mailslot Writes Failed | The total number of mailslot messages that have been successfully received, but that were unable to be written to the mailslot. |
| Browser\Missed Mailslot Datagrams | The number of mailslot datagrams that have been discarded due to configuration or allocation limits. |
| Browser\Missed Server Announcements | The number of server announcements that have been missed due to configuration or allocation limits. |
| Browser\Missed Server List Requests | The number of requests to retrieve a list of browser servers that were received by this workstation, but could not be processed. |
| Browser\Server Announce Allocations Failed/sec | The rate of server (or domain) announcements that have failed due to lack of memory. |

For information about the NTDS performance object that reports performance data for Active Directory, or about counters that report lightweight directory access protocol (LDAP) activity, see the Performance Counter Reference on the *Windows 2000 Resource Kit* companion CD.

### Troubleshooting Performance Problems with the Browser Service

Improving the performance of computers running the Browser service relies primarily on reducing traffic. You can do this in several ways:

- Configure the computer so that it does not send announcements to browsers on the domain and thereby reduce traffic. Do this by editing the registry. Add the **Hidden** registry entry to HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Services\lanmanserver\parameters with a data type of REG_DWORD and a value of 1 (hidden).

---

**Caution** Do not use a registry editor to edit the registry directly unless you have no alternative. The registry editors bypass the standard safeguards provided by administrative tools. These safeguards prevent you from entering conflicting settings or settings that are likely to degrade performance or damage your system. Editing the registry directly can have serious, unexpected consequences that can prevent the system from starting and require that you reinstall Windows. To configure or customize Windows, use the programs in Control Panel or Microsoft Management Console (MMC) whenever possible.

---

- Reduce the number of browser list entries. If a computer rarely shares network resources, configure it so that it doesn't become a browser server. This will reduce the size of the browse list that must be maintained and transferred upon request. Set the value of the **MaintainServerList** entry in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Browser\Parameters to No. For more information about browser service configuration, see "Browser Service" in the *Windows 2000 TCP/IP Core Networking Guide*.
- Eliminate unnecessary network protocols. If a network uses three protocols, all browser announcements and elections will be repeated three times, one for each protocol. If you can reduce the number of protocols, it will have a large impact on reducing browser-related network traffic.

In addition, two registry entries can be configured to control the amount of network traffic generated by the browser. Add the following entries to the registry subkey HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Browser\Parameters.

- **MasterPeriodicity**. **MasterPeriodicity** specifies how frequently a master browser contacts the domain master browser. The default value is 720 seconds (12 minutes), with a minimum of 300 seconds (five minutes), and a maximum value of 0x418937 (4,294,967 seconds). This entry has the REG_DWORD data type, and can be changed without restarting the computer. Setting the value of this entry too low can increase traffic on the WAN traffic.
- **BackupPeriodicity**. **BackupPeriodicity** specifies how frequently a backup browser contacts the master browser. Add this entry to the registry with the REG_DWORD data type. You must restart Windows 2000 to make changes to this entry effective. The default value for **BackupPeriodicity** is 720 seconds. Changing this value to 1800 (30 minutes) reduces the frequency of browse list updates. This entry does not affect the WAN, because backup browsers always communicate with a local master browser, never with a remote one.
- Limit the number of workgroups on the network.
- Configure a system to be a preferred master browser on each subnet where no domain controller exists.

### Redirector Object

Use the Redirector object counters for the Workstation service, and the Server and Server Work Queues objects for the Server service. The counters for these objects describe activity at the presentation layer of the networking architecture, as described in Table 9.3.

**Table 9.3 Redirector Object Counters**

| Counter | Comments |
|---|---|
| Redirector\Bytes Total/sec | The rate at which the Redirector is processing data bytes. This includes all application and file data in addition to protocol information such as packet headers. |
| Redirector\Current Commands | The number of requests to the Redirector that are currently queued for service. If this number is much larger than the number of network adapters installed in the computer, you might want to increase the maximum allowance for pending net commands in the **MaxCmds** registry in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet \Services\lanmanserver\parameters. The default value is 5. |
| Redirector\Network Errors/sec | Serious unexpected errors that generally indicate the Redirector and one or more servers are having serious communication difficulties. For example, a Server Message Block (SMB) Protocol error generates a network error. Look in the system event log for results. You might need to increase the value of the **SessTimeout** registry entry in HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Services\LanmanWorkstation \Parameters. The default is 45 seconds; values can range from 10 to 65535. |
| Redirector\Reads Denied/sec | The rate at which the server is unable to accommodate requests for Raw Reads. When a read is more than twice the negotiated buffer size of the server, the Redirector requests a raw read, which, if granted, would permit the transfer of the data without a lot of protocol overhead on each packet. To accomplish this, the server must lock out other requests, so the request is denied if the server is very busy. |
| Redirector\Server Sessions Hung | The number of active sessions that are timed out and unable to proceed due to a lack of response from the remote server. |
| Redirector\Writes Denied/sec | The rate at which the server is unable to accommodate requests for raw writes. When a write is much larger than the negotiated buffer size of the server, the Redirector requests a raw write, which, if granted, would permit the transfer of the data without a lot of protocol overhead on each packet. To accomplish this, the server must lock out other requests, so the request is denied if the server is very busy. |

### Server Object

The Server service supports file and print sharing and is important for communication between local and remote processes. Its companion, the Workstation service, provides network connections and communication. A computer uses the Workstation service to send requests to a server; the Server service responds to those requests. A server computer can run both services.

In general, memory and disk space are considerations on computers running the Server service, and overall server monitoring should include counters for these resources. Because many services might run on top of the Server service, you should also consider the requirements of those services when assessing server requirements. For more information about performance of the Server and Workstation services, monitor the computer using the counters in Table 9.4.

**Table 9.4 Server Object Counters**

| Object\counter | Description |
|---|---|
| Server\Blocking Requests Rejected | The number of times the server has rejected blocking SMBs due to insufficient count of free work items. Indicates whether the MaxWorkItem or MinFreeWorkItems server entries might need tuning. |
| Server\Bytes Total/sec | The number of bytes the server has sent to and received from the network. Provides an overall indication of how busy the server is. |
| Server\Context Blocks Queued/sec | The rate at which work context blocks had to be placed on the server file system process queue to await server action.<br><br>If this counter consistently averages higher than 50 milliseconds, the Server service is a bottleneck for all tasks on remote computers that are issuing remote I/O requests to the server. |
| Server\Errors System | The number of times an internal server error was detected. Unexpected errors usually indicate a problem with the server. |
| Server\Pool Nonpaged Failures | The number of times allocations from the nonpaged pool have failed. Indicates that the computer's physical memory is too small.<br><br>If this number consistently increases, the server is running out of the paged or nonpaged pool it originally allocated. If this occurs, you might want to consider increasing the resource. |
| Server\Pool Nonpaged Peak | The maximum number of bytes of nonpaged pool that the server has had in use at any one point. Indicates how much physical memory the computer should have. |

*(continued)*

**Table 9.4 Server Object Counters *(continued)***

| Object\counter | Description |
|---|---|
| Server\Pool Paged Failures | The number of times allocations from the paged pool have failed. Indicates that the computer's physical memory or page file are too small.<br>If this number consistently increases, the server is running out of the paged or nonpaged pool it originally allocated. If this occurs, you might want to consider increasing the resource. |
| Server\Pool Paged Peak | The maximum number of bytes of nonpaged pool that the server has had in use at any one point. Indicates how much physical memory the computer should have. |
| Server\Sessions Errored Out | Reports auto-disconnects along with errored-out sessions. To get a more accurate value for errored-out sessions, obtain the value for Server\Sessions Timed Out and reduce the Server\Sessions Errored Out value by that amount. |
| Server\Work Item Shortages | The number of times that no work item was available or could be allocated to service the incoming request. A work item is the location where the server stores an SMB. The amount available fluctuates between a minimum and maximum value that is configured based on how the server is configured and the amount of memory on the computer. If work item shortages are occurring, it might be caused by an overloaded server. If the Work Item Shortages counter value increases, consider changing the **Maxworkitems** registry entry in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters. The value can range from 1 to 512.<br>If the actual number of work items consistently matches the maximum set in the registry, the system consistently initiates flow control, which degrades performance. |

Some additional counters, although not performance related, provide useful information about server security. These include:

- Server\Errors Access Permissions
- Server\Errors Granted Access
- Server\Errors Logon

## Server Work Queues Object

The Server Work Queues performance object consists of counters that monitor the length of queues and objects in the queues. See Table 9.5.

**Table 9.5 Server Work Queues Object Counters**

| Counter | Description |
|---|---|
| Bytes Transferred/sec | The rate at which the server is sending and receiving bytes with the network clients on this CPU. Use to determine how busy the server is. |
| Queue Length | The current length of the server work queue for this CPU. A sustained queue length greater than four can indicate microprocessor congestion. This is an instantaneous count, not an average over time. |
| Total Bytes/sec | The rate at which the server is reading and writing data to and from the files for the clients on this CPU. Use this counter to determine how busy the server is. |
| Total Operations/sec | The rate at which the server is performing file read and file write operations for the clients on this CPU. This value will always be zero (0) in the Blocking Queue instance. Use this counter to determine how busy the server is. |
| Work Item Shortages | The number of times that an inadequate number of work items was available. A work item is a request to the server from a client; the server maintains a pool of available work items per CPU to speed processing. A sustained value greater than zero indicates the need to increase MaxWorkItems for the Server service. This value will always be zero (0) in the Blocking Queue instance. |

### Troubleshooting Problems Involving the Server Service

In some cases, the Server service can be associated with performance problems, as described in this section.

#### Event log reports an event ID 2009.

An event ID 2009 might appear listed in the event log if the server could not expand an internal table because the table had reached the maximum size. These internal tables track active sessions, resource connections, open files, and open searches, so this error message can be generated by problems involving these activities.

Table 9.6 lists counters you can use to investigate these activities and possible related problems.

**Table 9.6 Server Object Counters Used to Troubleshoot Event ID 2009 Events**

| Type of activity | Possible cause of event | Counters to monitor |
|---|---|---|
| Active sessions | No more user IDs (UIDs) exist to satisfy this SMB. This may be the result of maintaining unneeded user sessions on the server. This might include mapped drives in logon scripts or applications that automatically map drives to particular servers. | Redirector\Server Sessions<br>Redirector\Server Sessions Hung |
| Resource connections | No more free tree IDs (TIDs) exist to satisfy a TreeConnect SMB. | Redirector\Connects Core<br>Redirector\Connects Lan Manager 2.0<br>Redirector\Connects Lan Manager 2.1<br>Redirector\Connects Windows NT |
| Open files | No more file IDs (FIDs) could be allocated to process the various open file SMBs because of a shortage of available FIDs. | Server\Files Open<br>Server\Files Opened Total |

*(continued)*

**Table 9.6 Server Object Counters Used to Troubleshoot Event ID 2009 Events *(continued)***

| Type of activity | Possible cause of event | Counters to monitor |
|---|---|---|
| Open searches | Memory is allocated for search, find, and other SMB calls to store the current search state, but no additional memory could be allocated for storing search buffers. Because the Server service tends to allocate paged pool memory for storing search buffers, shortages of paged pool memory can cause this event. | Server\File Directory Searches<br>Memory\Pool Paged Bytes<br>Server Work Queues\Available Work Items |

MS-DOS applications or older applications that do not make Win32 calls do not have a method for closing searches after they complete. In order to handle this situation, the Server service uses several search time parameters to clear the search handle and reclaim the memory allocated to the search buffers. If you want to adjust the search time parameters to avoid events, change the values for the following entries in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters in the Windows 2000 registry:

| Subkey | Comments |
|---|---|
| MaxGlobalOpenSearch | The value of the **MaxGlobalOpenSearch** entry determines the maximum number of open searches allowed by the LanmanServer service. The default value is 4096 with a maximum value of 65536. To allocate more search handles, increase the value of **MaxGlobalOpenSearch** to 16,000 (decimal). |
| MaxKeepSearch | The value of the **MaxKeepSearch** entry determines the maximum amount of time in seconds that a search will remain open. The default value is 1800 seconds (30 minutes). Decrease the value of **MaxKeepSearch** to 900 seconds (15 minutes). |
| MinKeepSearch | The value of the **MinKeepSearch** entry determines the minimum amount of time in seconds that a search will remain open. The default value is 480 seconds (8 minutes). Decrease the value of **MinKeepSearch** to 240 seconds (4 minutes). |

**Caution** Do not use a registry editor to edit the registry directly unless you have no alternative. The registry editors bypass the standard safeguards provided by administrative tools. These safeguards prevent you from entering conflicting settings or settings that are likely to degrade performance or damage your system. Editing the registry directly can have serious, unexpected consequences that can prevent the system from starting and require that you reinstall Windows. To configure or customize Windows, use the programs in Control Panel or Microsoft Management Console (MMC) whenever possible.

# Trace Logs

The new trace log feature in Windows 2000 makes it possible to closely associate network input/output operations to the workload on a computer. Trace logs are a component of Performance Logs and Alerts. When configured to include Network TCP/IP activity, trace logs report TCP sends and receives as events, and measure the time required to complete these events, along with the bytes transmitted and other data. Using trace data, you can associate network traffic with the services or applications initiating it with the bandwidth utilization of the service or application. Note that trace logs are not output in readable format and must be parsed. The operating system does not provide a parser for trace logs. For information about creating and configuring trace logs, see Windows 2000 Help.

# Network Monitor

Unlike System Monitor, which is used to monitor anything from hardware to software, Network Monitor focuses exclusively on network activity. To understand the traffic and behavior of your network components, install and use Network Monitor.

## Network Monitor Features

Network administrators use Microsoft Windows 2000 Network Monitor to view and detect problems on local area networks (LANs). For example, as a network administrator, you can use Network Monitor to diagnose hardware and software problems when two or more computers cannot communicate. You can also copy a log of network activity into a file and then send the file to a professional network analyst or support organization.

Network application developers can use Network Monitor to monitor and debug network applications as they are developed.

Network Monitor monitors the network data stream, which consists of all information transferred over a network at any given time. Prior to transmission, this information is divided by the network software into smaller pieces, called frames or packets. Each frame contains:

- The source address of the computer that sent the message.
- The destination address of the computer that received the frame.
- Headers from each protocol used to send the frame.
- The data or a portion of the information being sent.

The process by which Network Monitor copies frames is referred to as capturing. You can use Network Monitor to capture all local network traffic or you can single out a subset of frames to be captured. You can also make a capture respond to events on your network. For example, you can make the network start an executable file when Network Monitor detects a particular set of conditions on the network.

After you have captured data, you can view it in the Network Monitor user interface. Network Monitor does much of the data analysis for you by translating the raw capture data into its logical frame structure.

For security reasons, Windows 2000 Network Monitor captures only those frames, including broadcast and multicast frames, sent to or from the local computer. Network Monitor also displays overall network segment statistics for broadcast frames, multicast frames, network utilization, total bytes received per second, and total frames received per second.

In addition, to help protect your network from unauthorized use of Network Monitor installations, Network Monitor can detect other installations of Network Monitor that are running on the local segment of your network. Network Monitor also detects all instances of the Network Monitor driver being used remotely (by either Network Monitor from Systems Management Server or the Network Segment object in System Monitor) to capture data on your network.

When Network Monitor detects other Network Monitor installations running on the network, it displays the following information:

- The name of the computer
- The name of the user logged on at the computer
- The state of Network Monitor on the remote computer (running, capturing, or transmitting)
- The adapter address of the remote computer
- The version number of Network Monitor on the remote computer

In some instances, your network architecture might prevent one installation of Network Monitor from detecting another. For example, if an installation is separated from yours by a router that does not forward multicasts, your installation cannot detect that installation.

Network Monitor uses a network driver interface specification (NDIS) feature to copy all frames it detects to its capture buffer, a resizable storage area in memory. The default size is 1 MB; you can adjust the size manually as needed. The buffer is a memory-mapped file and occupies disk space.

---

**Note** Because Network Monitor uses the local only mode of NDIS instead of promiscuous mode (in which the network adapter passes on all frames sent on the network), you can use Network Monitor even if your network adapter does not support promiscuous mode. Networking performance is not affected when you use an NDIS driver to capture frames. (Putting the network adapter in promiscuous mode can add 30 percent or more to the load on the CPU.)

---

## Installing Network Monitor

To set up Network Monitor, perform two steps:

- Install the Network Monitor driver on any computer from which you want to capture data for analysis with Network Monitor.
- Install the Network Monitor utilities on a computer running Windows 2000 Server on which data will be captured.

You can install the driver on a computer running either Windows 2000 Professional or Windows 2000 Server. Installing the driver also installs the Network Segment object for use in System Monitor.

Installing the driver does not install Network Monitor itself. Instead, install the Network Monitor Tools on a computer running Windows 2000 Server to install Network Monitor.

▶ **To install the Network Monitor driver**

1. Click **Start**, point to **Settings**, click **Control Panel**, and then double-click **Network and Dial-up Connections**.
2. In **Network and Dial-up Connections**, right-click **Local Area Connection**, and then click **Properties**.
3. In the **Local Area Connection Properties** dialog box, click **Install**.
4. In the **Select Network Component Type** dialog box, click **Protocol**, and then click **Add**.
5. In the **Select Network Protocol** dialog box, click **Network Monitor Driver**, and then click **OK**.

If prompted for additional files, insert your Windows 2000 CD, or type a path to the location of the files on a network.

To display and analyze captured data, use the following procedure to install Network Monitor Tools on a computer running Windows 2000 Server. Network Monitor Tools installs Network Monitor along with the Network Monitor driver. If you are running Windows 2000 Server and are installing Network Monitor Tools, you can bypass the preceding procedure; you do not need to install the Network Monitor driver separately.

▶ **To install Network Monitor Tools**

1. Click **Start**, point to **Settings**, click **Control Panel**, and then double-click **Add/Remove Programs**.
2. In the **Add/Remove Programs** dialog box, double-click **Add/Remove Windows Components**.
3. In the **Windows Component Wizard** dialog box, click **Next**.
4. Under **Components**, click **Management and Monitoring Tools**, and then click the **Details** button.
5. Under **Subcomponents of Management and Monitoring Tools**, select the **Network Monitor Tools** check box, and then click **OK**.
6. Click **Next** to proceed with installation, and then click **Finish** and **Close** to exit.

▶ **To start Network Monitor on a computer running Windows 2000 Server**

1. Click **Start**, point to **Programs**, and point to **Administrative Tools**.
2. Under **Administrative Tools**, click **Network Monitor**.

For information about how to work with the Network Monitor user interface, see Windows 2000 Server Help.

## Capturing Frame Data

When you've installed the Network Monitor driver on the computer from which to capture data (hereafter called the source computer) and installed Network Monitor Tools on the computer that will perform the capture (hereafter called destination computer), you can begin to capture data.

▶ **To capture data**

1. Open Network Monitor.
2. On the **Capture** menu, click **Start**.

   Or, click the **Capture** button on the toolbar.

As frames are captured from the network, statistics about the frames are displayed in the Network Monitor Capture window, as shown in Figure 9.2.

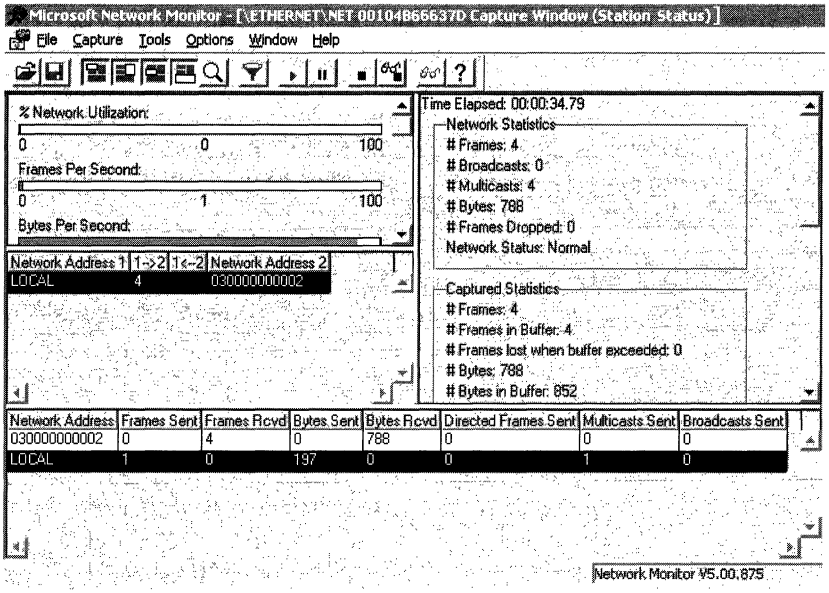



**Figure 9.2 Network Monitor Capture Window**

Network Monitor displays session statistics from the first 100 unique network sessions it detects. The Network Monitor Capture window includes the panes listed in Table 9.7.

**Table 9.7 Description of Display Options for the Capture Pane**

| Pane | Displays |
|---|---|
| Graph | A graphical representation of the activity currently taking place on the network. |
| Session Stats | Statistics about individual sessions currently taking place on the network. |
| Station Stats | Statistics about the sessions participated in by the computer running Network Monitor. |
| Total Stats | Summary statistics about the network activity detected since the capture process began. |

To reset statistics and see information on the next 100 network sessions detected, on the **Capture** menu, click **Clear Statistics**. To capture only those frames that originate with specific computers, determine the addresses of the computers on your network and associate the address with its DNS or NetBIOS name. After these associations are made, you can save the names to an address database (.adr) file that can be used to design capture filters and display filters. The capture filter allows you to specify criteria for inclusion in or exclusion from the capture. If the address is not available in the address database, try to capture all traffic and, after stopping and viewing the capture, use the **Find All Names** command on the **Display** menu to locate the address.

---

**Note** Capture filters can significantly increase the processor's workload because each packet must be processed through the filter and either saved or discarded. In some cases, using complex filters might result in missed frames.

---

An example of such a filter is an address pair, used to capture frames from specific computers on the network. An address pair consists of:

- The addresses of the computers between which you want to monitor traffic. Note that you can capture to a computer or to a router; however, you cannot select multiple address pairs with the **OR** operation. You must run multiple instances of Network Monitor to capture to either a computer or a router simultaneously. (An address is a hexadecimal number that identifies a computer uniquely on the network.)
- Arrows that specify the traffic direction you want to monitor.
- The INCLUDE or EXCLUDE keyword, indicating how Network Monitor should respond to a frame that meets a filter's specifications.

  Regardless of the sequence in which statements appear in the **Capture Filter** dialog box, EXCLUDE statements are evaluated first. Therefore, if a frame meets the criteria specified in an EXCLUDE statement in a filter containing both an EXCLUDE and INCLUDE statement, that frame is discarded. Network Monitor does not test that frame by INCLUDE statements to see if it meets that criterion also.

  For example, to capture all the traffic from Joe's computer *except* the traffic from Joe to Anne, use the following capture filter in the address section:

```
include   Joe <----> Any
exclude   Joe <----> Anne
```

  If there are no include lines, the default address
  *your_computer_name* – – – – Any
  is used by default.

Figure 9.3 shows the **Capture Filter** dialog box, accessed from the **Capture** menu or by pressing F8 in the Capture window.

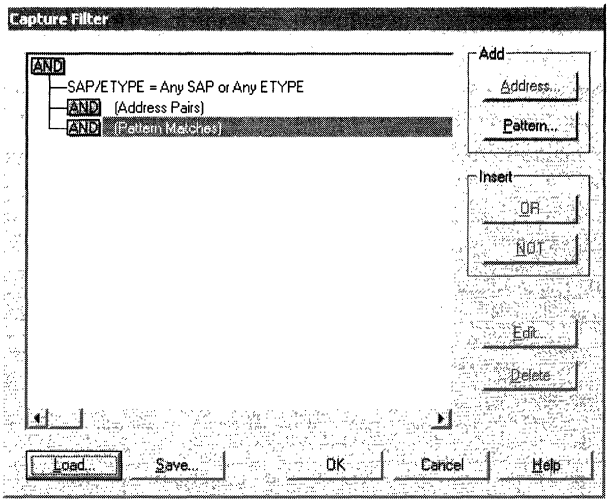



**Figure 9.3 Capture Filter Dialog Box**

To design a capture filter, specify decision statements in the **Capture Filter** dialog box. For information about display filters, see "Displaying Captured Data" later in this chapter.

By specifying a pattern match in a capture filter, you can:

- Limit a capture to only those frames containing a specific pattern of ASCII or hexadecimal data.
- Specify how many bytes into the frame the pattern must occur. This number of bytes is known as an offset.

  When you filter based on a pattern match, you must specify where the pattern occurs in the frame (how many bytes from the beginning or end). If your network medium has a variable size in the media access control protocol, such as Ethernet or Token Ring, specify to count from the end of the topology header.
- To capture frames sent using a specific protocol, specify the protocol on the capture filter SAP/ETYPE= line. Available protocols appear in the dialog box when you double-click the SAP/ETYPE= line. For example, to capture only IP frames, disable all protocols and then enable IP ETYPE 0x800 and IP SAP 0x6. By default, all of the protocols that Network Monitor supports are enabled.

- Use a capture trigger to automate actions to follow the capture. A trigger is a set of conditions that, when met, initiate an action. For example, before using Network Monitor to capture data from the network, you can set a trigger to stop the capture or to run a program or command file. You can also specify the conditions under which these actions will occur. One example of a trigger is a pattern match. You can save a trigger to the local computer if you save a capture filter. The default file path for saving filters is the \System32\Netmon\Captures directory in the root directory.

Table 9.8 describes the trigger types you can use to specify the condition that starts the trigger.

**Table 9.8 Trigger Types for Network Monitor Captures**

| Trigger type | Description |
|---|---|
| Nothing | No trigger is initiated. This is the default. |
| Pattern Match | Initiates the trigger when the specified pattern occurs in a captured frame. |
| Buffer Space | Initiates the trigger when a specified amount of the capture buffer is filled. |
| Pattern Match Then Buffer Space | Initiates the trigger when the pattern occurs and is followed by a specified percentage of the capture buffer being filled. |
| Buffer Space Then Pattern Match | Initiates the trigger when the specified percentage of the capture buffer fills and is followed by the occurrence of the pattern in a captured frame. |
| No Action | No action is taken when a trigger condition is met. This is the default. Even though you select **No Action**, the computer beeps when the trigger condition is met. |
| Stop Capture | Stops the capture process when the trigger condition is met. |
| Execute Command Line | Runs a program or batch file when a trigger condition is met. If you select this option, provide a command or the path to a program or batch file. |

If your computer uses multiple network adapters, use Network Monitor to collect data from multiple network adapters, and then either switch between the two adapters or run multiple instances of Network Monitor.

### ▶ To switch between adapters

- On the **Capture** menu, click **Networks**, and then select a different adapter.

Modem adapters appear as ETHERNET with a dial-up connection flag set to TRUE.

After capturing data, you might want to save it. For example, it is useful to save captures before starting another capture (to prevent loss of the captured data) if you think you might need to analyze the data later, or if you need to document network use or problems. When you save captured data, the data in the capture buffer is written to a capture (.cap) file.

## Displaying Captured Data

To simplify data analysis, Network Monitor interprets raw data collected during the capture and displays it in the Frame Viewer window.

To display captured information in the Frame Viewer window, from the **Capture** menu, click **Stop and View** while the capture is running. You can also display captures by opening a file with the .cap extension.

Figure 9.4 shows the key elements in the Frame Viewer window.

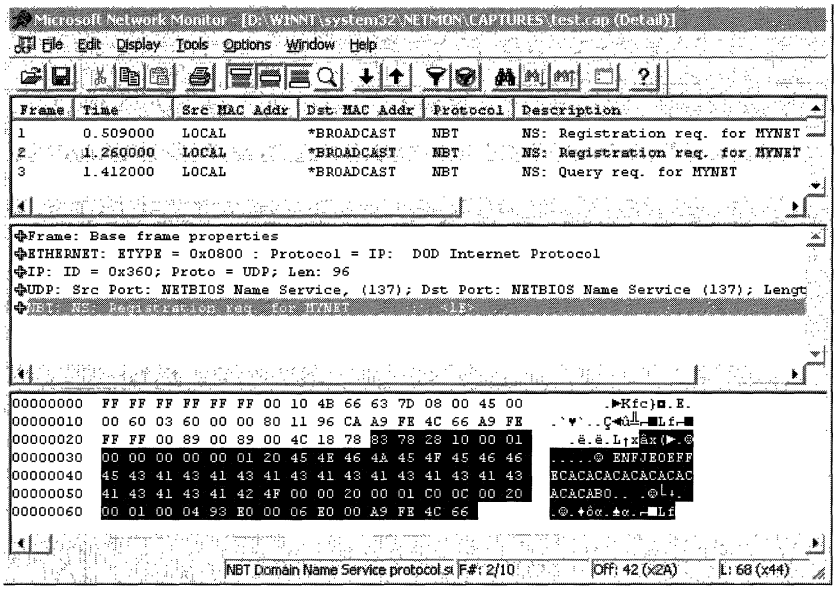



**Figure 9.4 Frame Viewer Window**

Table 9.9 lists Frame Viewer's panes.

**Table 9.9 Frame Viewer Panes**

| Pane | Displays |
|---|---|
| Summary | General information about captured frames in the order in which they were captured. |
| Detail | The parsed contents of the frame's data. |
| Hex | A hexadecimal and ASCII representation of the captured data. |

You can use a display filter to determine which frames to display. Like a capture filter, a display filter functions like a database query, allowing you to single out specific types of information. Because a display filter operates on data that has already been captured, it does not affect the contents of the Network Monitor capture buffer. You can filter a frame by:

- The frame's source or destination address.
- The protocols used to send the frame.
- The properties and values the frame contains. (A property is a data field within a protocol header. A protocol's properties, collectively, indicate the purpose of the protocol.)

Figure 9.5 shows the **Display Filter** dialog box, accessed from the **Display** menu or by pressing F8 in the Frame Viewer window.

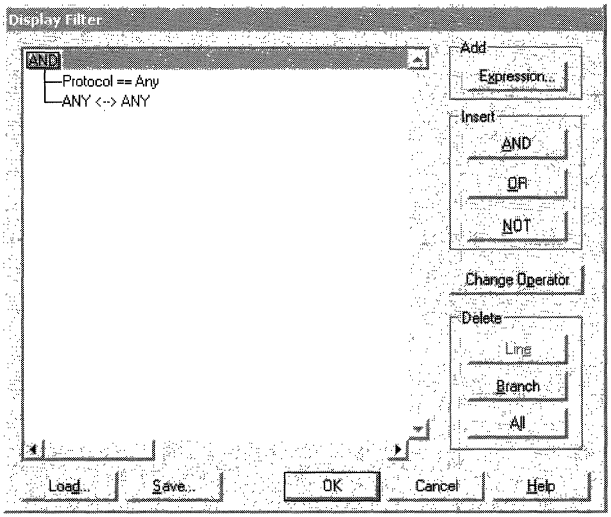



**Figure 9.5 Display Filter Dialog Box**

To design a display filter, specify decision statements in the **Display Filter** dialog box. Information in the **Display Filter** dialog box is in the form of a decision tree, which is a graphical representation of a filter's logic. When you modify display filter specifications, the decision tree reflects these modifications. Table 9.10 lists various types of filter items you can use.

**Table 9.10 Filter Item Options**

| Filter item | Description |
|---|---|
| Protocol | Specifies the protocols or protocol properties. |
| Address Filter (default is ANY <-- --> ANY) | Specifies the computer addresses on which you want to capture data. |
| Property | Specifies property instances that match your display criterion. |

You must click **OK** to save the specified decision statement and add it to the decision tree before adding another decision statement.

Although capture filters are limited to four address filter expressions, display filters are not. With display filters, you can also use **AND**, **OR**, and **NOT** logic.

When you display captured data, all available information about the captured frames appears in the Frame Viewer window. To display only those frames sent by a specific protocol, edit the Protocol line in the **Display Filter** dialog box.

Protocol properties are information that defines a protocol's purpose. Because the purpose of protocols varies, properties differ from one protocol to another.

Suppose, for example, that you have captured a large number of frames using the SMB protocol but want to examine only those frames in which the SMB protocol was used to create a directory on your computer. In this instance, you can single out frames where the SMB command property is equal to **make directory**.

When you display captured data, all addresses from which information was captured appear in the Frame Viewer window. To display only those frames originating from a specific computer, edit the ANY <-- --> ANY line in the **Display Filter** dialog box.

## Reviewing Captured Data

Perform the steps in the following list as part of your routine for reviewing and analyzing captured data:

- Follow a session using source and destination IP address and port numbers.
- If you find a Reset, focus on the sequence numbers and acknowledgments that precede it.
- Try to understand the activity you are seeing:
  - Is the sender doing retries?

    If so, note the number of retries and the time elapsed. The default number of retries for TCP/IP is 5. This value might be different for other protocols.

- Did the sender back up and resend the previous packet?
- Is the receiver asking for a missed frame by acknowledging a previous sequence number?
- Does the size of the data being sent and received correspond to the size of the maximum transmit unit (MTU) of the hardware? If not, you might have the wrong network settings
- Is there a lengthy delay for receipt of acknowledgements or for transmission of subsequent packets? This could indicate that the destination computer has inadequate resources or that the application is performing inefficiently.

A reset can be caused by time-outs at the TCP layer or by time-outs of higher-layer protocols. Resets originating at the TCP layer should be easy to read from the trace. It might be more difficult to determine the cause of resets originating from higher-layer protocols such as the server message block (SMB).

For example, an SMB read might time out in 45 seconds and cause a reset of the session even though communications are slow but working at the TCP layer. The trace might only narrow down what component is at fault. From there you might need to use other troubleshooting methods to determine the cause.

To see TCP sequencing when higher-level protocols are present, start Network Monitor and edit the Expression dialog box, using the following steps. Figure 9.6 shows the **Expression** dialog box.

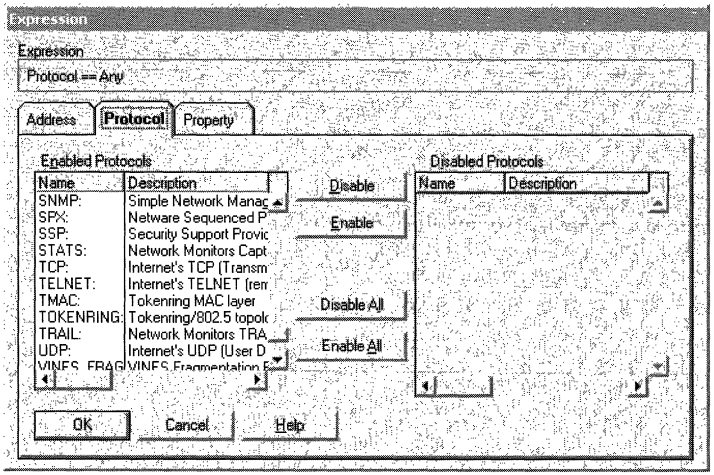



**Figure 9.6 Expression Dialog Box**

▶ **To see TCP sequencing**

1. Start Network Monitor.
2. Display captured data.
3. On the **Display** menu, click **Options**.
4. Select **Auto** (based on protocols in display filter), and then click **OK**.
5. Click **Display**, and then click **Filter**.
6. Double-click **Protocol=Any**.
7. Click the **Protocol** tab, and then click **Disable All**.
8. In the **Disabled Protocols** list box, click **TCP**.
9. Click **Enabled**, then click **OK**, and click **OK** again.

## Network Monitor Performance Issues

Network Monitor creates a memory-mapped file for its capture buffer. For best results, make sure to create a capture buffer large enough to accommodate the traffic you need.

In addition, although you cannot adjust the frame size, you can store only part of the frame, thus reducing the amount of wasted capture buffer space. For example, if you are interested only in the data in the frame header, set the frame size (in bytes) to the size of the header frame. Network Monitor discards the frame data as it stores frames in the capture buffer, thereby using less capture buffer space.

---

**Tip** Windows Event Viewer shows start, stop, and connection events for Network Monitor. To verify Network Monitor operation, or as a first step in tracking down Network Monitor problems, examine the event log.

---

# Resolving Network Bottlenecks

Typical causes for network bottlenecks are an overloaded server, an overloaded network, or a loss of network integrity. The following techniques can help to address some of these problems.

- If communicating over Token Ring, FDDI or switched Ethernet networks, attempt to balance network traffic by distributing client connections across multiple network adapters. When using multiple network adapters, make sure that the network adapters are distributed among the PCI buses. For example, if you have four network adapters with three PCI buses, one 64-bit and two 32-bit, allocate two network adapters to the 64-bit bus and one adapter to each 32-bit bus. Splitting the adapters across multiple Ethernet segments is an effective way to eliminate an overloaded network if the physical environment is switched Ethernet, and all adapters are on the same physical segment. For more information about adding network adapters, see "Adding Network Adapters" later in this chapter.
- Use adapters with the highest bandwidth available for best performance. Note that increasing bandwidth increases the number of transmissions that are taking place and in turn makes more work for your system, including more interrupts being generated. Remove unused network adapters to reduce overhead.
- Use adapters that support task offloading capabilities (checksum offloading, IPSEC offloading, and large send offloading).
- If your network uses multiple protocols, place each protocol on a different adapter. Make sure to use the most efficient protocols, especially ones that minimize broadcasts. Notice that reducing the number of protocols installed can increase performance.
- High rates of interrupts from network adapters can reduce performance. Using network adapters that batch interrupts by means of interrupt moderation can improve this performance problem, provided the adapter driver supports this capability. Another option is to bind interrupts arising from network adapters to a particular processor to improve performance. When using Interrupt Filter to bind a network adapter to a set of processors. For more information about processor affinity and Interrupt Filter, see "Measuring Multiprocessor System Activity" in this book. For more information about Interrupt Filter, see online Help on the *Windows 2000 Resource Kit* companion CD.
- Divide your network into multiple subnets or segments, attaching the server to each segment with a separate adapter. This reduces congestion at the server by spreading server requests.
- Although binding order is relevant on the client computer, there is no reason to reorder server bindings because the server accepts incoming connections based on the protocol used by the client computer.
- Use offline folders to work on network applications without being connected to a network. Offline folders make use of client-side caching, thereby reducing network traffic.

# Adding Network Adapters

You need to disable some bindings to allow network adapters in the same computer on the same network segment to operate correctly when using NetBEUI. The reason for this is that, if you have two adapters on the same segment and install NetBEUI, the computer sees duplicate computer names due to the resulting NetBIOS name conflict, generates an error, and fails to start system services correctly. This also occurs on bridged or switched networks. In this case you also see the Event ID 2505 error in the event log, indicating a duplicate name on the network. The error might appear in the event log as either of the following messages:

- Event ID : 2505 The server could not bind to the transport/device/netbt_ *adapter_driver* because another computer on the network has the same name. The server could not start.
- Event ID : 2505 The server could not bind to the transport/device/nbf_*adapter_driver* because another computer on the network has the same name. The server could not start.

This problem does not affect routable protocols such as TCP/IP or AppleTalk because when you install or reconfigure them, you have to choose a network adapter, and this automatically disables the protocol's bindings to other adapters. In addition, the operating system treats NetBT as a namespace and need not be disabled. You can manually configure multiple network adapters in a computer by disabling NetBEUI bindings to network adapters other than the one you want to use for the NetBEUI protocol. To do this, disable the following bindings:

- Server to NetBEUI protocol to *adapter_driver* to *adapter*
- Workstation to NetBEUI protocol to *adapter_driver* to *adapter*
- NetBEUI protocol to *adapter_driver* to *adapter*
- NetBIOS interface to NetBEUI protocol to *adapter_driver* to *adapter*

When you have disabled all of these bindings, network operations can proceed.

In addition, if all clients have equal network access to any of the network adapters on the multihomed computer, and all of the client and server network adapters are on the same subnet, you can help to distribute the client connections between the server network adapters by adding the **RandomAdapter** registry entry in **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Netbt\Parameters**. In this way, each network adapter will still respond to the name query, but each network adapter will choose the IP address randomly from all the network adapters on the server. This helps to distribute network sessions among the network adapters, but does not necessarily balance the load because network traffic might vary greatly between the sessions.

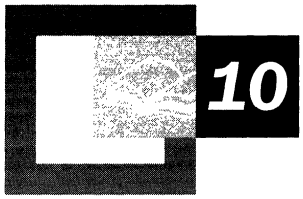
---

**Caution** Do not use a registry editor to edit the registry directly unless you have no alternative. The registry editors bypass the standard safeguards provided by administrative tools. These safeguards prevent you from entering conflicting settings or settings that are likely to degrade performance or damage your system. Editing the registry directly can have serious, unexpected consequences that can prevent the system from starting and require that you reinstall Windows 2000. To configure or customize Windows 2000, use the programs in Control Panel or Microsoft Management Console (MMC) whenever possible.

---

CHAPTER 10

# Measuring Multiprocessor System Activity




Managing the distribution of interrupt, thread, and deferred procedure call (DPC) activity for optimal performance means understanding factors that can limit and enhance the throughput on a symmetric multiprocessing (SMP) system.

**In This Chapter**

Overview of SMP Performance and Monitoring 431
Monitoring Activity on Multiprocessor Systems 433
Optimizing and Tuning Multiprocessor Installations 438
Application Design and Multiprocessor Performance 443
Network Load Balancing and Scaling 445

### Related Information in the Resource Kit

- For more information about general monitoring procedures, see "Overview of Performance Monitoring" in this book.
- For more information about monitoring single-processor systems, see "Analyzing Processor Activity" in this book.
- For more information about system problems that can affect performance, see "Troubleshooting Strategies" in this book.

# Overview of SMP Performance and Monitoring

Microsoft® Windows® 2000 is designed to implement symmetric multiprocessing (SMP). With symmetric multiprocessing, the operating system can run threads on any available processor. As a result, SMP makes it possible for applications to use multiple processors when additional processing power is required to increase the throughput capability of a system. Similarly, on Windows 2000, hardware interrupts and *deferred procedure calls* (DPCs), software interrupts at a low Interrupt Request Level (IRQL), can also run on any available processor determined by the hardware abstraction layer (HAL). Although most SMP systems running Windows 2000 dynamically distribute threads and hardware interrupts equally among all available processors, you might want to restrict threads and interrupts to one or more processors to improve processor cache locality and overall system throughput.

The following overview describes workloads that benefit most from scaling to an SMP system. This chapter also provides an overview of the steps involved in monitoring SMP systems and discusses the impact of SMP on system resources.

## Benefits of Scaling

The process of adding processors to your system to achieve higher throughput is called *scaling*. Applications that benefit from multiprocessor configurations are typically those that are processor-intensive, such as database servers, Web servers, and active file and print servers. Processor-intensive applications that use multiple processes or are multithreaded with asynchronous execution are also well suited to multiprocessor systems. Systems requiring heavy computation capability, including detailed calculation for scientific or financial applications, complex graphic rendering, computer aided design (CAD)–based modeling, or electrical-engineering design might also demand multiprocessor systems.

To understand the degree of benefit you can attain from scaling to multiple processors, determine the *scale factor*. The scale factor is a measurement of the increase in throughput you can expect to achieve. You can determine the scale factor of the system by comparing the throughput of one processor to the throughput of multiple processors.

The formula for computing scale factor is:

```
Number of transactions per second on n processors ÷ Number of
transactions per second on 1 processor
```

For example, if a single-processor system is using 100 percent of the processor handling 100 transactions per second and, with the addition of three processors, the resulting four-processor system can handle 320 transactions per second, the scale factor of the system is 3.2.

# Analyzing Performance on SMP Systems

You can monitor the activity of your SMP system by using the Performance console and its counters. To evaluate your system's performance, look at the following factors:

- Processor utilization and queue length. You might need to partition the workload so that a particular processor handles a particular workload to achieve better performance. For more information, see "Optimizing and Tuning Multiprocessor Installations" later in this chapter.
- Processor performance data, such as context switches, interrupts, threads, and processes. Activity rates and usage levels for these types of data that are higher than expected for a particular throughput can reveal inefficiencies in how your processors handle their workload. You might need to partition these types of activity. For more information, see "Optimizing and Tuning Multiprocessor Installations" later in this chapter.
- Overall resource utilization on your system. Scaling to multiple processors can increase the load on resources such as disks, memory, and network components, and it might be necessary to increase the capacity of these components. For more information, see "SMP Impact on System Resources" later in this chapter.

---

**Note** Application developers are in the best position to control how processes behave in an SMP environment. "Application Design and Multiprocessor Performance" later in this chapter provides guidelines for application developers who want to design programs that run well on multiprocessor systems.

---

# SMP Impact on System Resources

The increased processing power and throughput on SMP systems can cause other system resources such as memory, the system bus, disks, and network to have a heavier load. For example, arbitration among shared resources increases memory latency. This is because code running on SMP systems needs to lock shared data to ensure data integrity, and locking shared data might result in contention for shared data structures. Further, the synchronization mechanisms used to lock shared data structures can increase the processor code path. As a result, when the number of processors on a system is scaled up, it is generally necessary to increase other resources on the system, such as memory, disks, and network components, as described in the following.

**Processors.** A large processor cache delivers the best performance. In multiprocessor configurations, the cache can reduce memory latency on the shared system bus thereby reducing resource contention and access.

**Memory.** It is recommended that your SMP system have more than 64 MB of memory because it is likely that it carries a heavier workload than a single-processor system, and runs more processes and threads. In general, it is recommended to scale memory with processors. For example, if a single-processor system requires 64 MB of memory and a second processor is added to increase the throughput, double the memory to 128 MB. Although multiprocessor systems do require additional memory because of their per-processor data structures, the additional memory required is minimal compared to the typical demand of the working sets for the additional processes and threads running in a multiprocessing environment.

**Disk and network components.** When adding processors to the system it is generally necessary to increase the disk and network capacity of the system.

For information about the adequacy of resources in your configuration, see the earlier chapters under the "Performance Monitoring" section of this book.

# Monitoring Activity on Multiprocessor Systems

In addition to counters used to determine a baseline, as described in "Overview of Performance Monitoring" in this book, and processor-specific counters, as described in "Analyzing Processor Activity" in this book, the counters listed in Table 10.1 are useful for obtaining detailed information when evaluating the performance of multiprocessor systems.

**Table 10.1 Multiprocessor System Counters and Descriptions**

| Object(Instance)\Counter | Description |
|---|---|
| Process(*process_name*)\Thread Counter | Shows the last observed value, not an average. You need to monitor it at various times to get an accurate picture of activity. |
| Processor\% DPC Time | Determines how much time the processor is spending processing DPCs. DPCs originate when the processor performs tasks requiring immediate attention, and then defers the remainder of the task to be handled at lower priority. DPCs represent further processing of client requests. |
| Processor\% Interrupt Time | Determines how much time the processor is spending processing interrupts. Interrupts are generated when a client requests a connection or sends data.<br>If processor time is more than 90 percent and this value is greater than 15 percent, the processor is probably overburdened with interrupts. |

*(continued)*

**Table 10.1 Multiprocessor System Counters and Descriptions *(continued)***

| Object(Instance)\Counter | Description |
|---|---|
| Processor\DPCs Queued/sec | Monitors the rate at which DPCs are queued on a particular processor. (This counter does not measure the number of DPCs in the queue.) |
| Processor\Interrupts/sec | Reflects the rate at which the processor is handling interrupts. |
| System\Context Switches/sec | Indicates that the kernel has switched the thread it is running on a processor. A context switch occurs each time a new thread runs, and each time one thread takes over from another. A large number of threads is likely to increase the number of context switches. Context switches allow multiple threads to share time slices on the processors, but they also interrupt the processor and might reduce overall system performance, especially on multiprocessor computers. You should also observe the patterns of context switches over time. |
| System\System calls/sec | Monitors the frequency of calls to Windows 2000 system service routines. These are the services exported to applications from the kernel. |
| Thread(*process_name*\Thread #)\% Processor Time | Monitors processor time usage by threads on the system. |
| Thread\Context Switches/sec | Monitors context switches generated by individual threads. |

The following sections describe how to monitor and analyze the values reported in specific areas:

- Processor Time Data
- Processor Queue Lengths
- Interrupt and DPC Data
- Thread and Context Switching Data

# Examining Processor Time Data

The Processor\% Processor Time counter reports CPU utilization on your system. It is important to monitor this counter on SMP systems just as it is on single-processor systems. Observe processor usage patterns for individual processors and for all processors over an extended period. Also consider the number of threads in the system's processor queue to determine if high processor usage is limiting the system's ability to accomplish work.

Figure 10.1 depicts a high rate of processor use on a multiprocessor system.

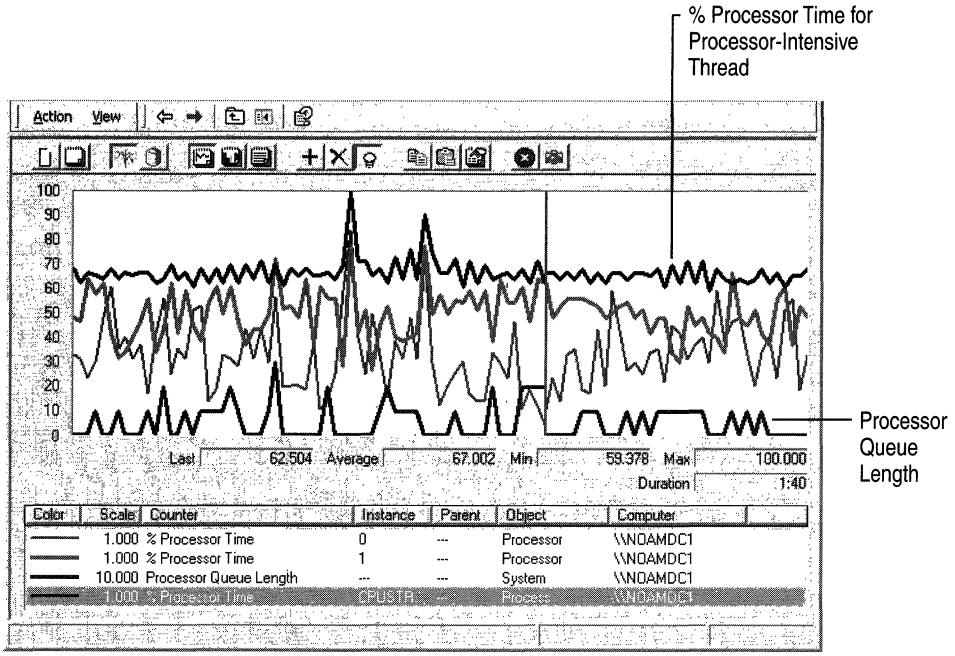



**Figure 10.1 Example of High Rate of Processor Usage on an SMP System**

On multiprocessor systems, the Processor\% Processor Time value reported by System Monitor will never exceed 100 percent for any particular processor or thread. On the other hand, the value of the % Processor Time reported for the Process object can report values over 100 percent; if such values occur, this could indicate that threads of the process are cumulatively using more than 100 percent of a processor. To get more detailed information, use the Thread object counters to analyze the processor time each thread within a process is using. Investigating other data described in this section, such as DPC activity or context switching, might help you to interpret high processor-time values.

## Observing Processor Queue Length

The System\Processor Queue Length counter is a rough indicator of the number of threads each processor is servicing. The processor queue length, sometimes called processor queue depth, reported by this counter is an instantaneous value that is representative only of a current snapshot of the processor, so it is necessary to observe this counter over a long period of time. Also, the System\Processor Queue Length counter is reporting a total queue length for all processors, not a length per processor.

The optimal processor queue length can vary based on processor utilization or other factors as follows:

- For busy systems that experience processor utilization in the 80 to 90 percent range and use thread scheduling, the queue length should range from one to three threads per processor. For example, on a four-processor (4P) system, the expected range of processor queue length on a system with high CPU activity is 4 to 12.
- On systems with lower CPU utilization, the processor queue length is typically 0 or 1.
- For systems running services that use fiber scheduling, such as Microsoft® SQL Server™ version 7.0, the typical processor queue length will range between 0 and 1, because there is a single thread on each processor that schedules fibers within the thread. (Fiber scheduling is enabled when the lightweight pooling option is selected.)

If the processor queue length exceeds the value recommended in the preceding list, it generally indicates that there are more threads than the current processor can service in an optimal way. Reducing the number of threads or providing more CPU power, either by adding processors or upgrading to faster processors, are optional methods of shortening the processor queue.

Figure 10.2 shows a long queue that has developed over time, as processors have been working at capacity.

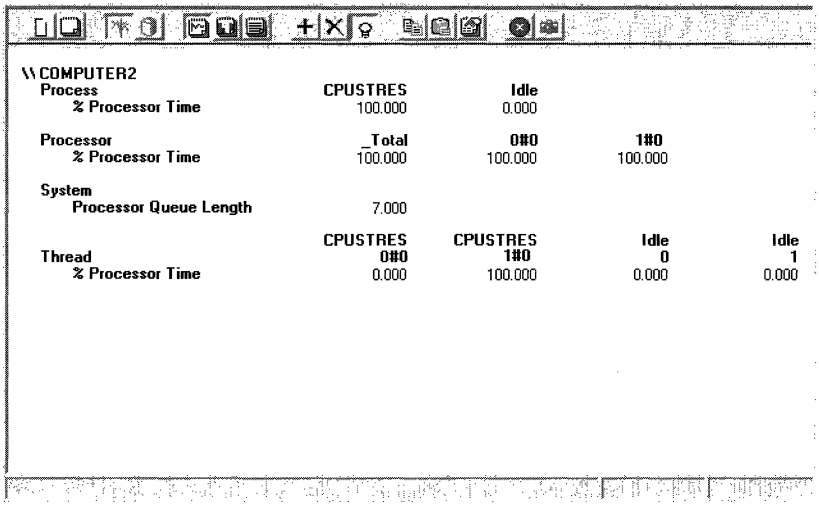



**Figure 10.2 Example of Long Queue at High Processor Usage**

When you begin to see longer queues, monitor additional counters for DPC and interrupt activity, as described in the following sections.

# Analyzing Interrupt and DPC Activity

On a multiprocessor system, interrupt-activity rates on different processors indicate how your system is distributing its workload among the available processors. Similar to context-switching rates, interrupt-activity rates can reveal distribution of work in a way that is inefficient and costly in terms of overall performance.

Most SMP systems can distribute hardware interrupts for handling on any of the processors. This allows interrupts to be handled by all processors, rather than concentrating the load on a single processor. In general, distributing interrupts provides better throughput. However, this depends greatly on the workload being processed.

When interrupts are distributed, the DPCs that arise from those hardware interrupts might run on different processors as well, meaning that their shared data is not cached but must be rewritten and reread. In addition, the assignment of a DPC to run on a different processor causes an *interprocessor interrupt* (IPI). An IPI is a high-Interrupt Request Level (IRQL) interrupt. Although it has a relatively low cost in terms of performance, like any interrupt, it reduces the efficiency of the processor cache of the target processor because cache lines are displaced by the interrupt.

At times when the system is developing a long processor queue or experiencing high rates of processor usage, observe the proportion of the processor's time that is spent servicing interrupts and DPCs. Compare the values of the Processor\% Interrupt Time and Processor\% DPC Time counters to Processor\% Processor Time.

If interrupt and DPC processor-time values are high, you need to investigate further:

- Observe the rate of interrupts and DPCs for each processor.
- Note whether interrupts and DPCs are distributed equally among all processors or whether one or more processors are servicing all of the interrupts or DPCs.

In general, a very high rate of interrupts might indicate a disk or a network adapter that needs upgrading or replacing. Test your components and rule out a hardware problem before proceeding. However, on a multiprocessor computer, the most common interrupt-related problem is its distribution among processors. It might be necessary to redistribute interrupts or DPCs or upgrade to a faster processor to avoid a bottleneck.

For information about how to manage interrupt, DPC, and other activity for better SMP performance, see "Optimizing and Tuning Multiprocessor Installations" later in this chapter.

## Monitoring Context Switches and Threads

When examining a multiprocessor system, monitor Thread\Context Switches/sec and Thread\% Processor Time to observe which threads are running on which processors, and to find out how frequently they switch between them to do the work of a particular process. This is important to know because there can be occasions when the switching of threads impedes optimal performance.

Typically, Windows 2000 uses a *soft affinity* algorithm that favors running a thread either on the last processor that serviced it or its *ideal processor,* a processor associated with a thread with a default value assigned by the system or optionally specified by the program developer in the application code. *Affinity* is the mechanism for associating threads of a process with a processor. With soft affinity, the association can vary between different processors; with hard affinity, the association is fixed to a set of one or more processors. In general, soft affinity is an optimal design that takes advantage of cache locality. However, when threads migrate from one processor to another processor, memory access to previously cached data may be slower. At worst, thread migration might cause false sharing because, to the cache coherency hardware of the CPU, it might appear that multiple processors are sharing a cache line.

Therefore, monitoring context-switching activity and thread processor usage is useful to application developers in understanding thread behavior.

# Optimizing and Tuning Multiprocessor Installations

One way to get maximum performance from a system is to partition your workload so that a specific processor is servicing all threads, interrupts, and DPCs associated with that workload. This avoids the inefficiencies introduced when these items are distributed among processors, resulting in frequent context switching and loss of cache locality. *Full partitioning*, that is, partitioning threads and interrupts by processor, is extremely effective in providing linear scaling on SMP systems. Depending on your workload, you can use a combination of the partitioning strategies or implement one strategy at a time.

Strategies for managing threads, interrupts, and DPCs for better overall SMP performance are described in the following sections:

- Thread Partitioning.
- DPC (Software Interrupt) Partitioning.

- Hardware Interrupt Partitioning.
- Bypassing I/O Counts.

**Note** When using server applications, such as Microsoft SQL Server, Microsoft® Exchange Server, or Microsoft® Internet Information Services, consult the documentation provided with those servers, which might have built-in optimizations.

## Thread Partitioning

Partitioning threads to specific processors is called setting a processor *affinity mask*. The affinity mask contains bits for each processor on the system, defining which processors a particular process or thread can use. When you set affinity for a process to a particular processor, all threads of the process inherit the affinity to the same processor.

Windows 2000 uses *soft* processor affinity, determining automatically which processor should service threads of a process. The soft affinity for a thread is the last processor on which the thread was run or the ideal processor of the thread. The Windows 2000 soft affinity thread scheduling algorithm enhances performance by improving the locality of reference. However, if the ideal or previous processor is busy, soft affinity allows the thread to run on other processors, allowing all processors to be used to capacity.

Windows 2000 also provides *hard affinity*, meaning that the processor affinity mask restricts the threads affected by the affinity mask to the processors specified by the mask. Threads restricted by a hard affinity mask will not run on processors that aren't included in the affinity mask. Hard affinity used with partitioning can improve performance of an SMP system substantially. However, be cautious when using hard affinity because it might cause the processors to have uneven loads. If processes that have had their affinity set to a specific processor are causing high CPU utilization on that processor while other processors on the system have excess processing capacity, the processes for which a hard affinity has been set might run slower because they cannot use the other processors.

If you want to ensure that a particular process or application doesn't have to share processing power with other tasks, you can use hard affinity to dedicate that process to one processor, leaving the remaining ones available for other work. This is easy to do using the **Set Affinity** command in Task Manager. For more information on Task Manager, see "Overview of Performance Monitoring" in this book.

Figure 10.3 shows the user interface for setting processor affinity.

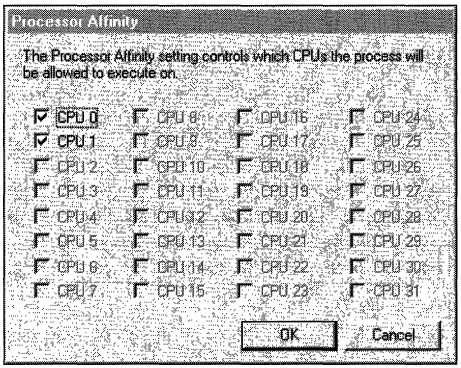



**Figure 10.3 Processor Affinity Dialog Box in Task Manager**

Another processor-partitioning option would be to divide discrete work items, such as tables in a database, among different processors. Consult the documentation accompanying your database software for information on partitioning databases for optimum performance.

---

**Note** Using functions in the Platform SDK, application developers who create applications using job objects can specify that all processes associated with the job object use the same processor affinity.

---

# DPC (Software Interrupt) Partitioning

In an environment with a large volume of network processing, you might want to control DPCs that arise from interrupts generated by your network adapters. The default handling of these DPCs might not allow you to attain optimal performance for your SMP configuration if you have a network-intensive workload.

Depending on the processor platform detected by the HAL, the operating system might configure different default settings to handle DPC distribution. If Windows 2000 detects that your system supports symmetric interrupt distribution (this includes Pentium Pro or later processors), the network DPCs associated with an interrupt remain on the processor that handled the interrupt. By doing so, the system improves processor cache locality and reduces interprocessor interrupts.

---

**Note** *I/O completion ports* are synchronization mechanisms defined in the Microsoft® Win32 API that are used with asynchronous I/O to limit the number of active threads that service I/O on the completion port. By limiting the number of active threads, I/O completion ports allow a pool of threads to be used such that the number of threads that share memory or hold a lock are minimized, thereby improving SMP scaling. The value defining that limit is called the concurrency value. When the concurrency value is set to 0, the completion port allows the number of active threads to be equal to the number of processors on a system. When there is one active thread per processor, there is no need for context switching or memory sharing. If a thread becomes blocked, the completion port allows another thread to be activated. On completion of a work item, the thread returns to the completion port for more work, thereby reducing context switching and improving locality of reference.

---

On a system with a relatively light network load, keeping the DPC processing on a single processor will improve the cache locality for the network DPCs and prevent network interrupts from occurring while a running thread has a lock on most of the processors. However, this distribution does not work well on systems with significant amounts of network traffic.

You can also add network adapters so that you have one network adapter for each processor. Generally, you should only add a network adapter if you also need the bandwidth, because each additional network adapter has some intrinsic overhead.

However, if one of the processors is nearly always active (if Processor\% Processor Time consistently equals 100 percent) and more than half of its time is spent servicing DPCs (that is, if Processor\% DPC Time exceeds 50 percent), then adding a network card is likely to improve system performance.

If you are adding or upgrading network adapters, choose adapters with drivers that support interrupt moderation or interrupt avoidance. Interrupt moderation allows a processor to process interrupts more efficiently by grouping several interrupts to a single hardware interrupt. Interrupt avoidance allows a processor to continue processing interrupts without new interrupts queued until all pending interrupts are complete.

If your network adapter does not use NDIS miniport drivers, you cannot modify the distribution of DPCs for better performance. For this reason, and because other NDIS optimizations might be unavailable, you might want to consider upgrading your network adapter.

---

**Note** By default, DPCs for disk interrupts remain on the processor that took the interrupt.

---

## Hardware Interrupt Partitioning

Some processor platforms can distribute interrupts across available processors; this capability is called *symmetric interrupt distribution*. These platforms typically include Pentium Pro and later processors; however, some Pentium processors also have this capability. Although symmetric interrupt distribution is designed as a way to balance interrupt activity, it can sometimes result in poor processor-cache performance. Partitioning interrupts to a specific processor is a strategy for addressing this problem. The Interrupt Filter tool on the *Windows 2000 Resource Kit* companion CD enables you to set the affinity for interrupts generated from disks or network adapters to a particular processor. This improves efficiency by taking advantage of cache locality, which can be lost when interrupts are serviced by any available processor. See Windows 2000 Professional Help for the Resource Kit tools for information about the requirements for using Interrupt Filter.

To help improve processor performance, some newer network adapter drivers provide an advanced feature known as *interrupt moderation* or *interrupt avoidance*. When the driver detects a high rate of interrupts from the network adapter, the interrupt-moderation code disables interrupts and accumulates the interrupts, sending them to the processor as a group of interrupts. Using a network adapter that supports interrupt moderation can provide better performance if your workload is network-intensive.

## Bypassing I/O Counts

By default, Task Manager continuously measures data for process I/O operations that you can select and display under the Processes tab in Task Manager. In a multiprocessor environment, this data is shared by the processors on which the process runs. When a process that generates considerable disk and network I/O, such as a database service, runs on several processors, updating the shared measurements of process I/O and global I/O operations can slow the system. You can improve the performance of I/O-intensive operations on SMP systems if you configure the system to bypass the global I/O counters and Task Manager process I/O counters. To do so, add the **CountOperations** entry to the registry as a REG_DWORD in HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\I/O System\. Set the entry value to 0. When so configured, Task Manager no longer provides per-process I/O measurements. For more information about Task Manager, see "Overview of Performance Monitoring" earlier in this book.

## Upgrading or Adding Processors

If you cannot improve the efficiency of your SMP configuration through the preceding steps, you might want to add or upgrade system processors. When doing so, choose processors with a large secondary (L2) cache. File server applications, Web servers, and databases are a few examples of the many workloads that benefit from a large processor cache. A large processor cache, ranging from 512K to 4MB or larger, is recommended to improve performance on active servers running Web servers or other connection-intensive operations.

# Application Design and Multiprocessor Performance

The design of applications and services has a significant influence on the performance of those programs in an SMP environment. This section is a brief summary of techniques that applications designers can implement to maximize the efficiency of their program on multiprocessing systems. More detailed discussions of writing applications that scale well to multiple processors appear on the Microsoft® Developer Network (MSDN).

Application developers can optimize application performance for SMP systems in the following ways:

- Keep the number of threads to a minimum. Generally, two to four application or server threads per processor works well.
- Limit processor queue depth. Keep the processor queue length (the number of ready threads waiting to run) in the range of two to three per processor. Depending on the characteristics of the application, such as the time a thread spends blocked or waiting for an I/O operation, the number of threads can be adjusted.
- Use a thread pool rather than one thread per client. It is more efficient to use a thread pool with an I/O completion port rather than to have a thread for each client, with the thread pool partitioned to each processor.
- Use I/O completion ports. I/O completion ports control the number of active threads to yield optimal throughput. Per-processor I/O completion ports can be implemented in an application or server to ensure completion of a work item from start to finish on the same processor.
- To minimize the cost of synchronization mechanisms, keep critical sections small and avoid shared data whenever possible. Synchronize shared data but do not try to synchronize code paths. Although critical sections—synchronization objects defined in the Win32 API— are a very fast method for mutual exclusion within a single multithreaded process, when contention arises, critical sections initiate context switching. Large numbers of critical section or spinlock acquisitions cause heavy data-access sharing and should be avoided.

  Spinlocks are an extension of IRQL on SMP systems. They are used to synchronize kernel and driver data structures among interrupts, DPCs, or threads of execution running concurrently on an SMP computer. A thread acquires a spinlock before accessing protected resources. The spinlock keeps other processors from accessing the critical section (shared data) until the spinlock is released. A processor that is waiting for the spinlock loops until the spinlock is released.

Another characteristic of spinlocks is the associated IRQL. Attempted acquisition of a spinlock temporarily raises the IRQL of the requesting processor to the IRQL associated with the spinlock. This prevents all lower IRQL activity on the same processor from running until IRQL is lowered. Interrupts at a higher IRQL can preempt the executing thread. In a driver, if IRQL is already at the desired level, use the spinlock acquire-and-release APIs that don't change IRQL. For more information about spinlocks and associated IRQLs, see the Driver Development Kit link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

- Avoid nested locks. Nesting locks can cause performance problems and reliability problems, such as deadlock. Always try to avoid nesting critical sections and spinlocks.
- Partition the workload, including interrupts. Whenever possible, partition the workload a server or application handles. Partitioning allows very effective use of system resources.

In addition, intensive memory access due to copying, zeroing (for C2 security), and checksum operations reduces the ability to scale effectively across multiple processors. You can identify some of these problems through profiling. For more information about profiling tools, see the Platform Software Development Kit (SDK) link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

- Change applications to prevent data that might be used concurrently by threads on different processors from residing in the same cache block, or use processor affinity to force use of updated cache blocks only on a single processor.
- Use asynchronous, overlapped I/O. In overlapped mode, a server application can initiate multiple I/O requests without waiting for previous requests to complete, thereby enabling it to service multiple clients asynchronously using a single thread.

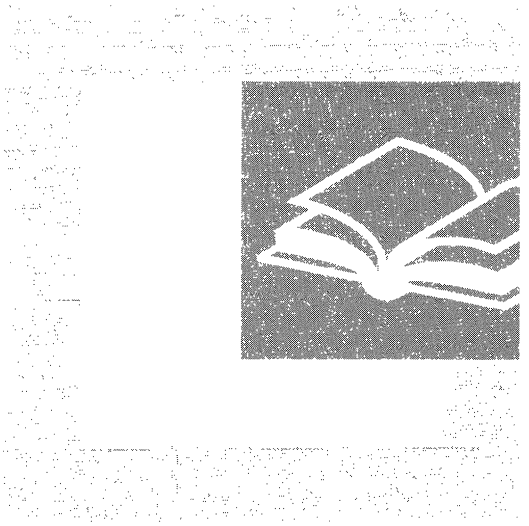
# Network Load Balancing and Scaling

Another scaling solution is provided by Network Load Balancing. With Network Load Balancing, two or more computers are combined to provide availability and scalability benefits to mission-critical applications.

A concept called *shared nothing* supports scalability in clusters. Clusters that use shared nothing have their own system bus and access to disks and networks. This reduces the sharing issues of SMP systems because additional resources are available without extra arbitration. In general, shared nothing is similar to partitioning in an SMP environment except that the resource access latency is lessened. However, if cluster members need to communicate, the latency of communication is much larger than in an SMP system. In addition, if sharing is necessary (that is, the cluster is not a shared-nothing environment), the cost of sharing in a cluster is much higher than the cost of sharing in an SMP system.

PART 3

# System Recovery



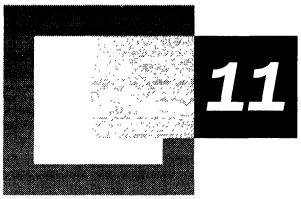
Preventative measures and strategic planning make computers running Microsoft® Windows® 2000 more failure resistant. Part 3 describes how to reduce the possibility of problems and recover from problems when they occur.

**In This Part**

Planning a Reliable Configuration 449
Backup 493
Repair, Recovery, and Restore 525

CHAPTER 11

# Planning a Reliable Configuration




Planning a configuration well facilitates data backup and security, maintenance and record-keeping, and the prevention or handling of contingencies, such as system failure or natural disaster. The following discussion includes realistic scenarios and checklists that you can adapt to your organization.

**In This Chapter**

Planning Considerations 451
Planning Tasks 457
Creating a Set of Windows 2000 Setup Floppy Disks 462
Creating a Windows 2000 Startup Floppy Disk 464
Planning a Fault-Tolerant Disk Configuration 465
Avoiding Single Points of Failure 485

### Related Information in the Resource Kit

- For more information about file systems, see "File Systems" in this book.
- For more information about recovery, see "Repair, Recovery, and Restore" in this book.
- For more information about security features and policies, see the *Microsoft® Windows® 2000 Server Resource Kit Distributed Systems Guide*.

# Planning Considerations

When planning a configuration that includes computers running Microsoft® Windows® 2000 Server and Microsoft® Windows® 2000 Professional, consider the following questions:

- How many computers running Windows 2000 Server and how many computers running Windows 2000 Professional do you need for your company?
- Are there computers running operating systems other than Windows 2000 Server and Windows 2000 Professional in your enterprise?
- Where do the computers need to be located, and how do they need to be connected?
- How many single points of failure can you tolerate in your enterprise?
- What methods can you implement for fault tolerance—the ability of a system to function when a computer component fails?
- What data on the computers needs to be backed up, and what is your backup strategy?
- What records do you need to maintain to troubleshoot problems or recover from failures?
- Have you developed and tested solutions for emergencies such as fire, flood, or power outage?

For each computer that uses Windows 2000 Server or Windows 2000 Professional, consider the following questions:

- What file systems are appropriate?
- How many controllers and disks are needed, and how do they need to be configured?
- Does some data need to be configured on fault-tolerant volumes?
  - If so, does fault-tolerant hardware need to be used on a computer running Windows 2000 Professional or Windows 2000 Server?
  - Or do you want to use the fault-tolerant features available with Windows 2000 Server?

### Climate Control

If the weather at the location of your company requires heating or cooling within the building to keep your computers and network devices within required operating temperatures, consider making the climate control system fault tolerant as well.

### Software Failure

Find a vendor who can provide the support you need in case of software failure. If your company does not already have a technical support group to assist users when there are software problems, form one.

### Power Control

Use an uninterruptible power supply (UPS) and battery backup to enhance fault tolerance. For more information about uninterruptible power supplies, see "Using an Uninterruptible Power Supply" later in this chapter.

## Approaches to Planning

Three proven approaches to planning include: top-down execution and responsibility, bottom-up execution and responsibility, and top-down policy and bottom-up planning and execution. Whatever approach you use, ongoing improvement and updates are better than total revision when new information, policies, or procedures occur.

**Top-down execution and responsibility** One department in the company is responsible for planning and coordinating policies and procedures. A centralized operations staff might be responsible for all or most computers running Windows 2000 and for executing backups. This method allows for creating a specialized group to develop procedures, procure and install hardware, conduct tests, and train personnel. Two disadvantages to this approach are that the specialists might not understand how data from the various departments relates to the overall business, and each department might not get appropriate attention.

**Bottom-up execution and responsibility** Each department is responsible for developing and implementing its own plans. Each department might also be responsible for the computers that its members use and for developing backup procedures. It is the department manager who best understands the impact of data loss. However, this approach can be costly because each department must train its own personnel and do its own research.

**Top-down policy and bottom-up planning and execution** One department develops the planning policy, which includes overall planning guidelines. Each department is responsible for developing and implementing its own plans and procedures. This method offers a consistent approach, which helps to ensure that each department knows the important details for a successful operation.

## System Maintenance

Routine, planned maintenance can prevent problems or minimize the effects when they occur. For example, if there are file system problems, you might not know about them until you restart the computer, when Chkdsk alerts you to the errors.

System administrators need to check both the system and application event logs daily. Impending file system problems can show up as errors in the system log before the file system is corrupted. Investigate any messages in the event log from FTDisk, Dmio, or disk device drivers.

Because disk fragmentation can cause performance problems, consider running a defragmentation program on a regular basis. You can use the Disk Defragmenter that is included with Windows 2000, or you can use a third-party defragmentation utility that runs under Windows 2000 and can defragment file allocation table (FAT) and NTFS file system volumes.

## Planning for Security

It is important to minimize the possibility of human error or deliberate sabotage. If anyone can access a computer running Windows 2000 and restart it, security software cannot protect your data from being damaged or stolen.

You can implement procedures that restrict physical access to your entire facility or to certain areas to which various personnel need access. Some ways to protect data include keeping computers in secure offices and password-protecting screen savers. Run virus checks on floppy disks before using them or disable floppy disk drives, which can sometimes be done by using Basic Input/Output System (BIOS) options. You can also run virus checks on a computer or physically disconnect it.

Windows 2000 allows you to create a profile for each user and restrict user access to files and servers, but no amount of planning can anticipate all the ways that data or computers can be damaged. Therefore, backup is a crucial part of your security and storage procedures. For more information about planning backup and storage procedures, see "Backup" in this book.

Windows 2000 includes auditing software that writes information to a security log whenever there is an attempt to breach security. You can log access to files and all logon attempts, valid or invalid. Your audit policy determines the types of events that are recorded in the security log. For more information, see Windows 2000 Server Help.

## Hardware and Software Upgrades

Even though you cannot avoid hardware and software upgrades, planning when and how to do them can minimize the time that an upgrade takes and can reduce the risk of failure.

When upgrading hardware or software keep the following in mind:

- Always back up important data files.
- Always back up the registry, master boot record (MBR), boot sector, and other critical system data.

- Maintain a current Emergency Repair Disk (ERD) for each computer.
- Maintain records about the computer configuration, such as jumper and DIP switch settings, interrupt requests (IRQs), and the hard disk configuration.

See the *Microsoft® Windows® 2000 Server Resource Kit Deployment Planning Guide* for more information about deploying Windows 2000.

# Contingency Planning

There are several topics that you need to keep in mind during your contingency planning.

## Disaster Planning

Just as every business needs to plan and budget for future growth, you need to plan for dealing with total or partial loss of business data. To determine what provisions to make for partial or complete loss of data, estimate the approximate cost in time and money to rebuild or replace critical data. Consider the following questions:

- Do you know the cost of reconstructing your company's financial, personnel, and other business data?
- Do you know if your business insurance would cover any or all of the cost of replacing data?
- Do you know how long it would take to reconstruct your business data? How might this affect future business?
- Do you know the cost per hour of computer downtime?

To prevent a natural disaster or sabotage from becoming a financial disaster for your business, test your plan for recovering and restoring critical data. Keep copies of your disaster recovery plan at on-site and off-site locations. Key personnel also need to keep a copy of critical data at home.

Because many books and magazine articles discuss disaster planning and recovery in detail, this section suggests topics to explore further instead of presenting detailed disaster plans to implement. Your insurance company can provide you with current and specific information for your situation.

The following are some important issues to consider when developing a comprehensive disaster plan to incorporate into your daily operations.

- What data do you need to back up, and how often do you need to do backups?
- What critical computer or other hardware configuration information, not saved during normal tape backups, needs to be saved?

- What data needs to be stored on-site or off-site, and how does it need to be stored?
- What training enables operators and administrators to respond quickly and effectively in an emergency?

## Assessing the Probability of Failure

Mean time between failures (MTBF) information supplied by some equipment manufacturers is generally only helpful if you do extensive analysis and modeling based on your company's pattern of use. Thus, it is recommended that MTBF information be used only as a relative measure of reliability.

Maintaining a record of past failures and their causes can be very helpful. This information can help you categorize failures by type, such as:

- Hardware failure on a server, client, or network component.
- Software failure of the operating system or applications on a server or client.
- Administrative error.
- User error.
- Deliberate damage, such as sabotage or a virus.

The following questions can help you analyze failures and your procedures for handling them:

- What was done or can be done to solve the problem?
- How long would or did the solution take?
- What would or did the solution cost?
- What actions have you taken to reduce the recurrence of each recorded failure?
- What changes have you made that might affect the number of failures? Changes might include the size of local area networks (LANs) or wide area networks (WANs), or the number of:
  - Servers
  - Clients
  - Users
  - Administrators
  - Intermediary devices
  - External connections

## Estimating Replacement Costs

There are several ways to measure the costs of recovering from problems. Some are easy to calculate, such as:

- Replacing file servers, mail servers, or print servers.
- Replacing servers running applications such as Microsoft® SQL Server™ or the Microsoft® Systems Management Server.
- Replacing gateway servers running Routing and Remote Access, Microsoft® SNA Server, Proxy Service, or Novell NetWare.
- Replacing workstations for personnel.
- Replacing computer components, such as hard disks and network adapters.
- Replacing products that have a set shelf life.

Far more difficult to measure, but just as devastating, are the invisible costs of computer downtime, such as lost sales, lost customer goodwill, lost productivity, increased costs for makeup time, missed contractual obligations, and loss of competitiveness.

If you have kept records of failures, you might find them useful in your contingency planning. You can investigate ways to avoid each failure, or to minimize the downtime associated with the failure. If you have cost information for the failures, you can then compare the cost of each failure to the cost of preventing or minimizing the failure. Table 11.1 describes two examples of failure, the costs related to each failure, and the effects of implementing solutions or workarounds to avoid future failures.

**Table 11.1 Examples of the Effects of Failure**

| Category | Example One | Example Two |
|---|---|---|
| Failure description | File server in sales department down, network adapter failure | Router failure between development and testing department |
| Effect | Lost sales | Lost productivity of employees |
| Total downtime last year | Three hours | 16 hours |
| Costs of failure per hour | $10,000 | Average hourly wage of 10 affected employees is $18/hr |
| Annual downtime costs | $30,000 | $2,880 |
| Possible resolution or workaround | Three spare network adapters at $100 each | Put an alternate router in place or obtain a spare router |

*(continued)*

**Table 11.1 Examples of the Effects of Failure** *(continued)*

| Category | Example One | Example Two |
|---|---|---|
| Expected costs of resolution or workaround | $300 | $500–$2,000 |
| Estimated savings during first year with resolution in place | $29,700 | $880–$2,380 |

# Planning Tasks

Maintaining information about your hardware and software configurations, developing plans to deal with specific types of failure, and training personnel to handle recovery tasks can help minimize the effects on your systems and reduce costs incurred from downtime.

## Maintaining Configuration and System Information

Hardware failures, power failures, and human errors can prevent Windows 2000 from starting successfully. Recovery is easier if you know the configuration of each computer and its history and if you back up critical system files when making changes to your Windows 2000 configuration.

**Important** If you have a clustered system and a disaster occurs, you need to recreate your disk structures and signatures. Write down this information and store it safely. If this information is not recorded, it cannot be recovered.

You can also use the DumpConfig command-line utility (Dumpconfig.exe) to output a computer's disk and volume configuration information to the command-line window or to a text file. In the event of disk failure, you can use this information to restore disk, partition, and volume configurations. When the Disk Management snap-in is displayed, you can record your disk configuration by printing the display or saving it to a file.

It is strongly recommended that you create a technical reference library for all hardware and software documentation. This library can also include other documentation related to your computers that you want to have in a central location. Useful materials to include in this library include:

- Vendor documentation, including manuals, receipts, warranties, proofs of purchase, and so on.
- Insurance policy and any claims.
- Information about any kits, tools, and add-ons that you have installed.

- All internal documentation that you have generated, including policies, procedures, and training guides.
- Hardware configuration information for each computer, including:
  - Location, cabling, intermediary devices, and external connections.
  - Computer type, and the models and serial numbers of system components.
  - Computer BIOS manufacturer, revision level, and BIOS modifications.
  - CMOS information.
  - For each component, information such as IRQ, direct memory access (DMA) addresses, and the input/output (I/O) port.
  - Amount of parity or nonparity random access memory (RAM).
  - Information about the video display and the video card.
  - Complete configuration of the disk subsystem, including the make, model, and serial number of each disk and controller, and their type, such as Small Computer System Interface (SCSI) or Enhanced Integrated Device Electronics (EIDE).
  - Configuration information for other storage media, such as tape, optical disk, or removable disk.
  - Types and configuration of network adapters.
  - Configuration information about any other adapters, such as multiple port adapters.
- Software configuration information and backups for each computer, including:
  - Disk map, noting location, type, and size of each partition, logical drive, and dynamic volume.
  - System state data backups.
  - The Windows 2000 Setup floppy disks created from the Windows 2000 Setup CD which can be used to start a disabled system.
  - Windows 2000 versions installed and the partitions on which they are installed.
  - User information, including which users are allocated disk space and what utilities and application programs they are using.
  - Applications and the volumes on which they are installed.
  - Licensing information.
  - Any service packs installed.
  - All hotfixes installed.
  - All Software Support Disks installed, when applicable.

- Updated contact information, including:
  - Appropriate personnel to respond to problems or emergencies.
  - Administrators.
  - Vendors and consultants.
  - Managers.
  - Critical users.

## Developing a Recovery Plan

The skill and experience of support personnel is crucial in getting failed systems back online with minimal disruption to your business. They need to be trained to troubleshoot problems and to implement recovery procedures when problems occur.

In preparing a recovery plan, start by imagining some typical scenarios. Your plan needs to answer the following questions:

- Does each operator know how to restart the computer when the disk containing the operating system fails?
- Do you have a set of Windows 2000 Setup floppy disks that you can use to start a disabled Windows 2000 computer? Do you have a Windows 2000 startup floppy disk?
- When was the last time you tested the Windows 2000 Setup floppy disks or the Windows 2000 startup floppy disk?
- If a controller fails, how long will it take to replace the controller? Is the hardware configuration information immediately available?
- If a hard-disk drive fails, do you know how to replace it? If the drive was part of a fault-tolerant disk set, can you replace the disk and quickly fix the mirrored or RAID-5 volume?

Efficient recovery from system failures requires practice. Schedule drills several times a year that simulate computer crashes and disk failures.

Computers that have recently been taken out of service or are being prepared for production service can be used for training, or you can configure computers specifically for testing and training. Use training sessions and drills to update and document recovery procedures.

## Testing Your System for Possible Problems

Testing is an important component of your contingency planning. You can use testing to try to predict failure situations and to practice recovery procedures. Be sure to stress test all functionality.

The following list identifies some of the failures that you need to test:

- Individual computer components, such as hard disks, controllers, processors, and RAM.
- External components such as routers, bridges, switches, cables, and connectors.

The following are some useful situations to simulate in your stress tests:

- Heavy network loads.
- Heavy disk I/O to the same disk.
- Heavy use of file, print, and applications servers.
- Large number of users simultaneously logging on.

## Testing Recovery Procedures

Once you have created a set of Windows 2000 Setup floppy disks, a Windows 2000 startup floppy disk, and an ERD, and have backed up the system state data, use the floppy disks, ERD, safe mode and the Recovery Console to practice recovering from problems. This can help you to be diligent about making backups of the system state data and user data. This can also help you determine how long these procedures take to accomplish.

Your testing needs to help you determine the best recovery procedure for a particular situation. Determine when to use the set of Windows 2000 Setup floppy disks, the Windows 2000 startup floppy disk, safe mode, and the Recovery Console to restart your computer and when to use the ERD and Backup to replace files.

Your test computer needs to allow you to conduct the following tests:

- Look at the MBRs, partition tables, and boot sectors.
- Find the backup boot sector on an NTFS partition.
- Deliberately destroy and recover MBRs and boot sectors.
- Delete Windows 2000 system files and restore them by using the ERD.

Be sure to test recovery procedures before bringing a new computer or server into production. Every operator needs to have both primary and refresher training in recovering from the most common causes of unexpected downtime. Testing needs to include:

- Testing the UPS on the computer running Windows 2000 Server and on hubs, routers, and other network components.
- Testing the disaster plan.
- Restoring from your backups.
- Testing your ability to rebuild mirrored and RAID-5 volumes if you are running Windows 2000 Server and using software fault-tolerant volumes (a mirrored volume or a RAID-5 volume) or using hardware RAID arrays.

If a network adapter or other network component fails on the domain controller, the server operator needs to be familiar with the procedure for promoting a member server to be a domain controller, and demoting the failed server. Someone who is familiar with the procedure for reinstalling and reconfiguring the network adapter also needs to be available.

If a data volume fails, the operator must be able to restore the data from backup quickly and efficiently. The restore procedure needs to be tested frequently, both to ensure the skill of the operator and to test the quality of the backup tapes. The only way to test the quality of backup tapes is to do a full restore, which guarantees that the data is up-to-date and of consistent quality.

If your backup procedures involve the use of other computers running Windows 2000 Server or Windows 2000 Professional, verify that those backup and restore procedures work as expected.

## Documenting Recovery Procedures

You need to develop step-by-step procedures for recovering from a variety of potential failures. You can use these procedures for:

- Testing a new computer before putting the computer into a production environment.
- Training new administrators and operators.
- Creating an operations handbook, including procedures for setting up new user accounts, conducting backups, maintaining ERDs, and completing other common administrative tasks.

Update your documentation when you make configuration changes to your computers or network, especially when you install a new operating system or change the utilities that you use to maintain your system.

## Training Personnel for Recovery

Properly trained personnel can reduce the likelihood of failures and can reduce their severity when they do occur.

Effective training must start with the basics. Administrators and operators need to have a good understanding of Windows 2000 Server and Windows 2000 Professional. The Microsoft Certified Professional Program is a good starting point.

The following are several training and technical support options that can help you prepare to handle problems:

- Access Microsoft and vendor information through the Internet.
- Subscribe to TechNet.
- Have support personnel complete self-study programs.
- Enroll support personnel in vendor-approved or third-party courses.
- Enroll support personnel in certification courses in the use and troubleshooting of system hardware and software.
- Create a technical library that is available for personnel.
- Develop your own training courses.

You can also contract with Microsoft, hardware vendors, and third-party consultants for support. For more information about technical support options available to Microsoft customers, see "Troubleshooting Strategies" in this book.

# Creating a Set of Windows 2000 Setup Floppy Disks

To prepare for the possibility of a system failure on a computer that cannot be started from the CD-ROM drive, you need to create a set of four Windows 2000 Setup floppy disks that you can use to start the computer.

Use the Windows 2000 Setup floppy disks under the following circumstances: starting setup, starting the Recovery Console, and starting the Emergency Repair Process. However, before deciding that a computer must be started from a CD-ROM or floppy disks, try starting the computer in safe mode. After you have started the disabled computer from the Windows 2000 Setup floppy disks, you can use either the Recovery Console or the ERD, if you have prepared one. For more information about safe mode, the Recovery Console and the Emergency Repair Process, see "Startup Process" and "Repair, Recovery, and Restore" in this book.

You can create floppy disks for starting a disabled system by using the Windows 2000 Setup CD on any computer that is running Windows 2000. You need four blank, formatted, 3.5-inch, 1.44-MB floppy disks. Label them as follows:

- Windows 2000 Setup Boot Disk
- Windows 2000 Setup Disk #2
- Windows 2000 Setup Disk #3
- Windows 2000 Setup Disk #4

The Windows 2000 Setup floppy disks must match the operating system on the computer that you want to start. You cannot use Windows 2000 Setup floppy disks created from the Windows 2000 Professional Setup CD to start a computer that is running Windows 2000 Server.

---

**Note** Recreate your Windows 2000 Setup floppy disks after installing any Windows 2000 service packs.

---

▶ **To create the Windows 2000 Setup floppy disks**

1. Insert a blank, formatted 1.44-MB disk into the floppy disk drive on a computer that is running Windows 2000.
2. Insert the Windows 2000 Server Setup CD.
3. Click **Start**, and then click **Run**.
4. In **Open**, type:

   **d:\bootdisk\makebt32 a:**

   Where **d:** is the drive letter assigned to your CD-ROM drive, and then click **OK**.
5. Follow the screen prompts.

---

**Note** Makebt32.exe runs under Windows 2000, Microsoft® Windows NT® version 4.0 and Microsoft® Windows NT® version 3.51. If you are using a computer that is running Microsoft® MS-DOS®, Microsoft® Windows®98, or Microsoft® Windows® 95 to create the Setup floppy disks, in Step 4, type:

**d:\bootdisk\makeboot a:**

---

# Creating a Windows 2000 Startup Floppy Disk

You can use the Windows 2000 startup floppy disk to access a drive with a faulty startup sequence. This disk can access a drive that has either the NTFS, FAT16, or FAT32 file system installed. The Windows 2000 startup floppy disk can help with the following startup problems:

- Corrupted boot sector.
- Corrupted MBR.
- Virus infections.
- Missing or corrupt NTLDR or Ntdetect.com.
- Incorrect Ntbootdd.sys.

The Windows 2000 startup floppy disk can also be used to start from the surviving member of a mirrored volume. However, it might be necessary to modify the file Boot.ini first.

The Windows 2000 startup floppy disk cannot be used for the following problems:

- Incorrect or corrupted device drivers that have been installed into the Windows 2000 System directory.
- Startup problems that occur after the boot loader starts.

Use the emergency repair process to work around or fix these problems. On the **Windows 2000 Advanced Options** menu, choose **Last Known Good Configuration** or, if necessary, reinstall Windows 2000. For more information about the emergency repair process, see "Repair, Recovery, and Restore" in this book. For more information about the Last Known Good Configuration, see "Startup Process" in this book.

---

**Note** The Windows 2000 startup floppy disk must include the files NTLDR, Ntdetect.com, Boot.ini, and the correct device driver for your hard disk drive. NTLDR, Ntdetect.com, and Boot.ini usually have their file attributes set to system, hidden, and read-only. You do not need to reset these attributes for the Windows 2000 Startup disk to work properly.

---

▶ **To create a Windows 2000 startup floppy disk**

1. Insert a blank, formatted 1.44-MB disk into the floppy disk drive on a computer that is running Windows 2000.
2. Copy NTLDR, Ntdetect.com, and Boot.ini from the Windows 2000 Setup CD to the floppy disk.

3. If you have a SCSI system and the SCSI BIOS is not enabled, copy Ntbootdd.sys to the floppy disk.

**Important** Recreate your Windows 2000 startup floppy disk after the installation of any Windows 2000 service pack.

If you are making a Windows 2000 startup floppy disk while your computer is still functional, copy Boot.ini from the hard disk drive. If you have not yet made a Windows 2000 startup floppy disk and you are encountering problems, you must get the files from another Windows 2000 computer and modify Boot.ini to match the configuration of the computer that is having problems.

For more information about creating and using a Windows 2000 startup floppy disk, see the Microsoft Knowledge Base link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

# Planning a Fault-Tolerant Disk Configuration

A redundant array of independent disks (RAID) is a fault-tolerant disk configuration in which part of the physical storage capacity contains redundant information about data stored on the disks. The redundant information is either parity information (in the case of a RAID-5 volume), or a complete, separate copy of the data (in the case of a mirrored volume). The redundant information enables regeneration of the data if one of the disks or the access path to it fails, or a sector on the disk cannot be read. Windows 2000 Server implements these fault-tolerant configurations in its software. Use Disk Management, shown in Figure 11.1, to configure mirrored volumes and RAID-5 volumes, and to reconstruct the volume when there has been a failure.

**Note** You can also use hardware RAID arrays.

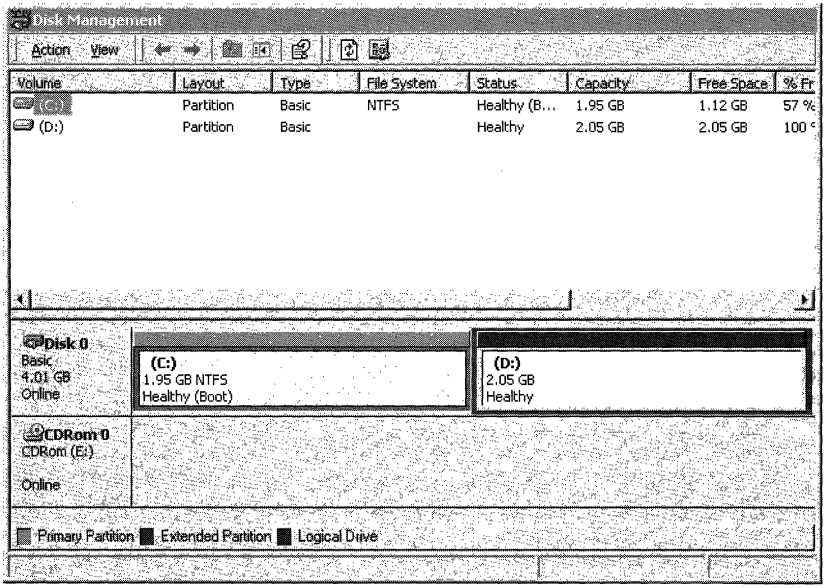



**Figure 11.1 Disk Management**

Table 11.2 shows the volume status descriptions that appear in the graphical view of the volume and in the Status column of the volume in List view.

**Table 11.2 Disk Management Volume Status Descriptions**

| Volume Status | Description |
|---|---|
| Healthy | The volume is accessible and has no known problems. This is the normal volume status. No user action is required. Both dynamic volumes and basic volumes display the Healthy status. |
| Healthy (At Risk) | The dynamic volume is currently accessible, but I/O errors have been detected on the underlying disk. If an I/O error is detected on any part of a disk, all volumes on the disk display the Healthy (At Risk) status. Only dynamic volumes display the Healthy (At Risk) status. |
| Initializing | The dynamic volume is being initialized. Only dynamic volumes display the Initializing status. |
| Resynching | The volume's mirrors are being resynchronized so that volume mirrors contain identical data. |
| Regenerating | Data and parity are being regenerated for the RAID-5 volume. |

*(continued)*

**Table 11.2 Disk Management Volume Status Descriptions *(continued)***

| Volume Status | Description |
|---|---|
| Failed Redundancy | The data on the volume is no longer fault tolerant because one of the underlying disks is not online. The Failed Redundancy status applies only to mirrored or RAID-5 volumes. |
| Failed Redundancy (At Risk) | The data on the volume is no longer fault tolerant and I/O errors have been detected on the underlying disk. If an I/O error is detected on any part of a disk, all volumes on the disk display the (At Risk) status. Only dynamic mirrored or RAID-5 volumes display the Failed Redundancy (At Risk) status. |
| Failed | The volume cannot be started automatically. Both dynamic and basic volumes display the Failed Status. |
| Missing | A dynamic disk is corrupted, powered down, or disconnected. |

Certain disk subsystems implement RAID technology completely within the hardware. Some of these hardware implementations support hot swapping of disks, which enables you to replace a failed disk while the computer is still running Windows 2000 Server. A RAID array is a fault-tolerant disk subsystem where all of the fault tolerance is implemented by the hardware. For information about RAID arrays that are compatible with Windows 2000, see the Windows 2000 Hardware Compatibility List (HCL) link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

All RAID disk configurations regenerate data to satisfy read requests when a disk or a path to a disk has failed. Regeneration involves reading data from other disks in the volume. RAID disk configurations also reconstruct the missing data onto the new disk when you have replaced the failed disk (or the path to it). When you have implemented a fault-tolerant volume, use Disk Management reconstruct the data. If you are using a RAID array, see the vendor's documentation for information about reconstructing data.

Configuring mirrored volumes and RAID-5 volumes is discussed later in this chapter. For more information about recovering from failures of mirrored volumes and RAID-5 volumes, see "Repair, Recovery, and Restore" in this book.

## Dynamic Disks and Disk Groups

Windows 2000 uses a new disk layout that extends beyond disk partitioning. New to Windows 2000 are dynamic disks and Disk Groups. Dynamic disks allow more flexibility during configuration than basic disks. For example, dynamic disks allow you to increase the size of a volume or add a mirrored volume without restarting the computer. A Disk Group is a collection of disks managed as a collection that helps you organize dynamic disks and helps prevent data loss.

Disk Groups are unique to dynamic disks. Each disk in a Disk Group stores replicas of the same configuration data. This configuration data is stored in a 1-megabyte (MB) region at the end of each dynamic disk. Since this information is contained on each disk, you can move them to another computer or install another disk without losing this information. All dynamic disks in a computer are members of the same Disk Group and have the following characteristics:

- They overcome partitioning rules from Windows NT 4.0.
- They extend or create fault-tolerant sets without the need to restart the computer.
- They are not registry dependent: that is, disk information for both basic and dynamic disks and fault tolerance information for dynamic disks is kept on the disk in Windows 2000.

---

**Note** To simplify use with docking stations, you cannot use dynamic disks on portable computers. Also, you cannot use dynamic disks on removable media. This simplifies movement of media between machines.

---

## Converting Basic Disks to Dynamic Disks

In Windows 2000, you can convert a basic disk to a dynamic disk. When you convert a disk, any existing partitions or fault tolerance structures on the disk are checked and then the disk is assigned a Disk Group identity and a copy of the current Disk Group configuration. Windows 2000 also adds dynamic volumes to the configuration.

---

**Note** By default, when you add a disk to an existing computer, it is considered a new disk. New disks are always dynamic disks.

---

## Moving Disks

You can move basic and dynamic disks from one computer to another. For both basic and dynamic disks, you need to physically move the disk from the computer, and then either restart the computer or use the **Rescan Disks** command on the **Action** menu of Disk Management.

When you remove a dynamic disk from a computer, the remaining online dynamic disks retain information about it and its volumes. The removed disk is displayed in Disk Management as a **Dynamic/Offline** disk and assigned the status of "Missing."

You need at least one online dynamic disk to retain information about Missing disks and their volumes. When you remove the last dynamic disk, you lose the information, and the Missing disks are no longer displayed in Disk Management.

**Note** When you move basic fault-tolerant sets from a computer running Windows NT 4.0, you need to save the configuration to a floppy disk, and then use Disk Management to restore the hard disk configuration. It is strongly recommended that you move all disks from the fault-tolerant set.

## Connecting New Disks to a Computer

After you connect the disks to the new computer, from the Disk Management snap-in, select **Rescan Disks** on the **Action** menu. The New Signature Wizard is displayed. When you physically connect a dynamic disk, the rescan causes the disk to be displayed in Disk Management as **Dynamic/Foreign**. You might need to convert it to **Dynamic/Online**, and then activate the volumes. When you connect a new disk to a computer, or rescan the dynamic mirrored volume and move only one disk, it displays as **Dynamic/Offline**, and you need to manually activate it.

**Note** When you connect a basic disk with existing partitions, you must select **Rescan Disks** from the Disk Management snap-in.

## Importing Foreign Disks

If you move one or more disks from a Disk Group to another computer that contains its own Disk Group, the Disk Group you moved is marked as Foreign until you import it into the existing group.

To use Foreign/Dynamic disks, use the Import Foreign Disks operation associated with one of the disks. The manual operation lists one or more Disk Groups, identified by the name of the computer where they were created. If you expand the details on a Disk Group, it lists the locally-connected disks that are members. Click the appropriate Disk Group, and then click **OK**. You can then view the dialog box that lists volumes that were found in the Disk Group, along with some indication of the status of those volumes.

Since volumes can span multiple disks using simple disk spanning, striping, mirroring, or RAID-5 redundancy mechanisms, the display status of a volume in the **Import Foreign Disks** dialog box can become complicated if not all of the disks have been moved. Another complication can arise from moving a disk, and then later moving additional disks. This is supported, but can be complicated. For this reason, all fault tolerant and non-fault-tolerant volumes that span disks should be moved at the same time.

The state of a volume after import depends on whether the volume is simple, mirrored, RAID-5, or spans multiple disks (simple striping behaves like spanning in this respect). It also depends on if the whole or part of the volume is moved, and if the volume is moved incrementally. The state depends on whether changes to the configuration of a partially moved volume were made on the original or the new computer:

- When all disks that contain parts of a volume are concurrently moved from one computer to another, the state of the volume after the import is identical to the original state. All simple volumes on any moved disks are recovered to their original state.
- On a non-redundant volume that spans multiple disks, if only some disks are moved from one system to another, the volume is disabled during import: it also becomes disabled on the original system. As long as the volume is not deleted on either the original or the target system, the remaining disks can be moved later. When all disks are finally moved over, the volume is recovered to its original state.
- RAID-5 volumes can remain online if they are missing one disk. The status of the volume is displayed as either Failed Redundancy or Failed Redundancy (At Risk). If a disk is moved to a different computer, all or all but one of the disks must be moved. If all disks are moved, the status of the volume cycles from Regenerating to Healthy. If all but one of the disks are moved, the status of the volume is Failed Redundancy (At Risk). Whether the volume remains online depends on whether the data is known to be valid or can be regenerated from the parity and the data. Parity starts out as invalid when a RAID-5 volume is first created, since the parity blocks must be computed, which takes time. Parity is also marked as invalid after a system crash, because an in-progress write can leave a discrepancy between parity blocks and the corresponding data blocks. If the parity of a RAID-5 volume is valid, one disk can be missing and the RAID-5 volume still becomes (or remains) online. If parity is not valid, then all parts of the RAID-5 volume need to be available for the volume to become (or remain) online.
- If both halves of a mirrored volume are moved as a set, the volume functions normally and your data continues to be fault tolerant. If only one half of a mirrored volume is moved to another computer, and then you return it to the original computer, the two halves of the original mirrored volume do not function as a mirrored volume. To restore the mirrored volume, break the original mirrored volume and then recreate it.

- If one up-to-date mirror is moved first, the mirror on the resulting Missing disk (for the non-moved mirror) can be removed and reallocated to another disk. This leaves a fully mirrored volume on the target computer. In this case, if the second original mirror is moved over, it conflicts in a way that cannot be resolved readily. When this happens, the second mirror comes over as a new volume.

**Warning** Note that when removing and moving disks with mirrored volumes, if one of the volumes becomes damaged during the move, your data is no longer fault tolerant.

It is better to remove all disks at the same time, and to add all disks at the same time. With SCSI disks, this is fairly easy: stop using the disks, and then defer the Rescan disks request until all disks are removed.

With any kind of disk, turn off the original system before removing the disks, and then turn off the target system before adding the disks.

For information about disk groups, see the Knowledge Base link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

## Simple Volumes

Simple volumes are the dynamic disk equivalent of primary partitions in Windows NT 4.0 and earlier versions. When you have only one dynamic disk, you can only create a simple volume. Simple volumes can be created on dynamic disks only. They cannot contain partitions or logical drives, and they can only be accessed by Windows 2000.

You can increase the size a simple volume to include unallocated space on the disk, but the volume must be formatted with the version of NTFS used in Windows 2000. You cannot increase the size of a simple volume that was converted from a basic disk partition. Increasing the size of a simple volume to include space on other disks of the same computer creates a spanned volume.

## Spanned Volumes

Spanned volumes are created from unallocated space from between two and 32 dynamic disks. The areas of unallocated space used to create spanned volumes can be of different sizes. You can increase the size of a spanned volume, but the volume must be formatted with NTFS.

After a spanned volume is extended, no portion of it can be deleted without deleting the entire spanned volume. Spanned volumes cannot be mirrored or striped and do not offer fault tolerance. If one of the disks containing a spanned volume fails, the entire volume fails.

**Note** Only Windows 2000 can recognize a spanned volume. On a multiple-boot computer, spanned volumes are unusable by other operating systems.

## Striped Volumes

Windows 2000 Server and Windows 2000 Professional provide software support for striped volumes, which are configured by using Disk Management. Striped volumes improve I/O performance by distributing I/O requests across disks. Striped volumes are composed of stripes of data of equal size written across each disk in the volume. They are created from equally sized, unallocated areas on up to 32 physical disks. For Windows 2000, the size of each stripe is 64 kilobytes (KB).

Conceptually, a striped volume is similar to a table in a document, where a disk is a column and a stripe is one of the entries in the table. A stripe includes all of the entries in one row. Table 11.3 illustrates the structure of a striped volume and shows the order in which data is written to the striped volume.

**Table 11.3 Structure of a Striped Volume**

| Stripe Number | Disk 1 | Disk 2 | Disk 3 | Disk 4 |
|---|---|---|---|---|
| Stripe 1 | 1 | 2 | 3 | 4 |
| Stripe 2 | 5 | 6 | 7 | 8 |
| Stripe 3 | 9 | 10 | 11 | 12 |
| Stripe 4 | 13 | 14 | 15 | 16 |
| Stripe 5 | 17 | 18 | 19 | 20 |

In the preceding table, stripe 1 consists of the four stripes that are the first block on each of the four disks. Stripe 5 is made up of the stripes that are the last block on each disk.

When you write data to a striped volume, the data is written *across* the stripes in the volume. Thus, using Table 11.3 as an example, a file of 325 KB could occupy the following space:

- 64 KB on stripe 1 of disk 1
- 64 KB on stripe 1 of disk 2
- 64 KB on stripe 1 of disk 3

- 64 KB on stripe 1 of disk 4
- 64 KB on stripe 2 of disk 1
- 5 KB on stripe 2 of disk 2

The physical disks in a striped volume do not need to be identical, but there must be unused space available on each disk that you want to include in the volume. You cannot increase the size of a striped volume after it is created. To change the size of a striped volume, you must first complete the following steps:

- Back up the data.
- Delete the striped volume by using Disk Management.
- Create a new, larger, striped volume by using Disk Management.
- Restore the data to the new striped volume.

Striped volumes do not contain redundant information. Therefore, the cost per megabyte on a striped volume is identical to that for the same amount of storage configured from a contiguous area on a single disk. If one disk fails, the whole striped volume fails and no data can be recovered. The reliability for the striped volume is less than the least reliable disk in the set.

Stripe sets are used for performance reasons. In general, striped volumes work well when you need to distribute I/O operations. Access to the data on a striped volume is usually faster than access to the same data would be on a single disk, because the I/O is spread across more than one disk. Therefore, Windows 2000 can be seeking on more than one disk at the same time, and can have simultaneous read or write operations occurring.

A striped volume works well in the following situations:

- When users need rapid read access to large databases or other data structures.
- When storing program images, dynamic-link libraries (DLLs), or run-time libraries for rapid loading. Operating systems such as Windows 2000 that use memory mapped images can benefit from using striped volumes.
- When collecting data from external sources at very high transfer rates. This is especially useful when collection is done asynchronously.
- When multiple independent applications require access to data stored on the striped volume. When the operating system supports asynchronous multithreading, which helps load balance disk read and write operations.

# Mirrored Volumes

A mirrored volume provides an identical twin of the selected volume. All data written to the mirrored volume is written to both volumes, which results in disk capacity of only 50 percent.

**Note** In this chapter, the terms original disk and original volume refer to the original volume that the data was written to, and shadow disk and shadow volume refers to the disk or volume that contains the copy.

Figure 11.2 shows a mirrored volume. To a user, only one read or write occurs to satisfy a request for data. For each user read request, Dmio.sys creates one read. For each user write request, Dmio.sys creates two writes.

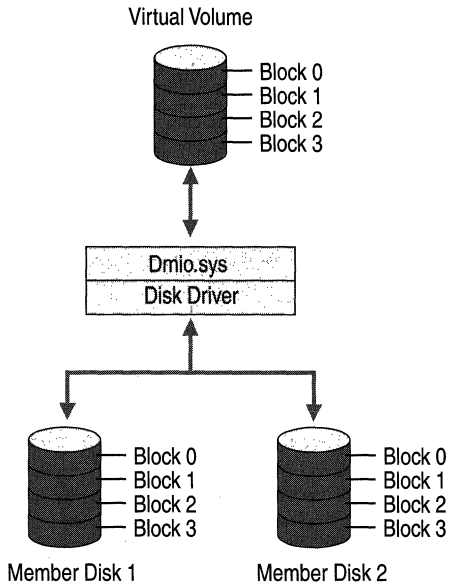



**Figure 11.2 Mirrored Volume**

If there is a read failure on one of the disks, Dmio reads the data from the other disk in the mirrored volume. If there is a write failure on one of the disks in the mirrored volume, the remaining disk is used for all accesses.

Because dual-write operations can degrade system performance, many mirrored volume configurations use duplexing, where each disk in the mirrored volume resides on its own disk controller.

Any volume can be mirrored, including the system and boot volumes. The disk that you select for the shadow volume does not need to be identical to the original disk in size, or in its number of tracks and cylinders. This means that you do not have to replace a failed disk with an identical model. The unused area that you select for the shadow volume cannot be smaller than the original volume. If the area that you select for the shadow volume is larger than the original, the extra space on the shadow disk can be configured as another volume.

**Note** As with striped volumes, you cannot add disk space to a mirrored volume to increase the size of the volume later.

When compared to a RAID-5 volume, a mirrored volume implementation:

- Has a lower entry cost because it requires only two disks, whereas a RAID-5 volume requires three or more disks, but does cost more per gigabyte.
- Requires less system memory.
- Provides very good overall performance.
- Does not show performance degradation during a failure. However, in the event of a single write error, redundancy is lost.
- Has a higher cost-per-megabyte.

**Note** The term system volume refers to the disk volume containing hardware-specific files needed to start Windows 2000 (such as the files NTLDR and Boot.ini). The boot volume contains the operating system files and support files. The boot volume and the system volume can be the same volume.

A mirrored volume works well in the following situations:

- When extremely high data reliability is required. A duplexed mirrored volume has the best data reliability because the entire I/O subsystem is duplicated.
- When simplicity is important. Mirrored volumes are simple to understand and easy to set up.

You might not want to use a mirrored volume if cost is a critical factor. Mirrored volumes are the most expensive solution based on the cost per unit of data storage.

## Advantages of Mirrored Volumes

Random disk read operations on mirrored volumes are more efficient than on a single volume. Dmio has the capacity to load balance read operations across the physical disks. With current SCSI technology, two disk read operations can be done simultaneously.

Recovery from a disk failure is very rapid. Mirrored volumes offer the fastest data recovery, with the least impact on system performance, because the shadow volume contains all of the data. There is no data recomputation building needed to restore the system. When you configure your boot volume on a mirrored volume, you do not have to reinstall the Windows 2000 Server to be able to restart the computer.

---

**Note** If a mirrored system fails, you do not need to restart the computer. The system continues to run and, if your hardware supports it, you can replace the failed drive then resynchronize the mirror. Resynchronization is the process by which a mirrored volume's mirrors are made to contain identical data. During resynchronization, performance is affected because the computer is performing a lot of I/O operations.

---

It is not necessary to use identical physical disks or to have the same volumes on each disk, although it is recommended that you use identical disks if you put your system volume on a mirrored volume. However, sufficient unused space on the second disk is required to create the shadow volume.

### Disadvantages of Mirrored Volumes

Disk write operations are less efficient. Because data must be written to both disks, there is a slight performance penalty. However, the penalty is not 100 percent. In many situations, a user-mode application is not affected by the extra disk update.

Mirrored volumes are the least efficient in terms of space utilization. Because the data is duplicated, the space requirements for a mirrored volume are higher than for a RAID-5 volume.

For information about recovering from the failure of a mirrored volume, see "Repair, Recovery, and Restore" in this book.

## RAID-5 Volumes

A RAID-5 volume dedicates the equivalent of the space of one disk in the RAID-5 volume for storing the parity stripes, but distributes the parity stripes across all the disks in the group. The data and parity information are arranged on the volume so that they are always on different disks.

Table 11.4 shows the order in which data is written to a RAID-5 volume that consists of five stripes on five disks.

**Table 11.4 Structure of a RAID-5 Volume**

| Stripe Number | Disk 1 | Disk 2 | Disk 3 | Disk 4 | Disk 5 |
|---|---|---|---|---|---|
| Stripe 1 | parity 1 | 1 | 2 | 3 | 4 |
| Stripe 2 | 5 | parity 2 | 6 | 7 | 8 |
| Stripe 3 | 9 | 10 | parity 3 | 11 | 12 |
| Stripe 4 | 13 | 14 | 15 | parity 4 | 16 |
| Stripe 5 | 17 | 18 | 19 | 20 | parity 5 |

The parity stripe is the exclusive OR (XOR) of all the data values for the data stripes in the stripe. If no disks in the RAID-5 volume have failed, the new parity for a write can be calculated without having to read the corresponding stripes from the other data disks. Thus, only two disks are involved in a write operation: the target data disk and the disk that contains the parity stripe. Figure 11.3 shows the steps that are involved in writing data to a RAID-5 volume.

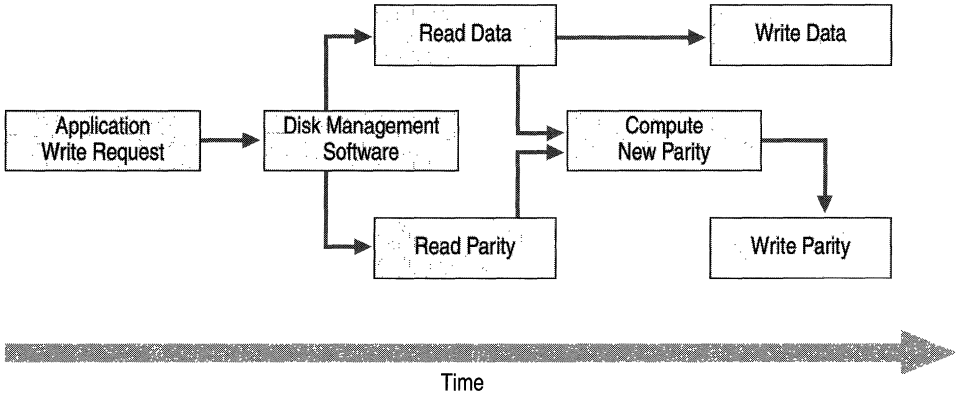



**Figure 11.3 Writing Data to a RAID-5 Volume**

Implementing a RAID-5 volume requires a minimum of three and a maximum of 32 disks in the set. The physical disks do not need to be identical. However, there must be equal size blocks of unused space available on each physical disk in the set. The disks can be on the same or different controllers. As with striped volumes, you cannot add disks to a RAID-5 volume if you need to increase the size of the volume later.

If one of the disks in a RAID-5 volume fails, none of the data is lost. When a read operation requires data from the failed disk, the system reads all of the remaining good data stripes in the stripe and the parity stripe. Each data stripe is subtracted (with XOR) from the parity stripe; the order is not important. The result is the missing data stripe.

When the system needs to write a data stripe to a disk that has failed, it reads the other data stripes and the parity stripe and backs them out of the parity stripe, leaving the missing data stripe. The modifications needed to the parity stripe can now be calculated and made. Because the data stripe is unavailable, it is not written; only the parity stripe is written.

There is no effect on a read operation when the disk that failed contains a parity stripe. (The parity stripe isn't needed for a read, unless there is a failure in a data stripe.) When the failed disk contains a parity stripe, the system does not compute or write the parity stripe when there is a change in a data stripe.

## Advantages of RAID-5 Volumes

RAID-5 volumes are well suited for large sequential reads. Typically, RAID-5 is used in database mining. A RAID-5 volume also works well in the following situations:

- In large databases where reads occur much more often than writes. Performance degrades as the percentage of write operations increases. Database applications that read randomly work well with the built-in load balancing of a RAID-5 volume.
- Where a high degree of fault tolerance is required without the expense (incurred by the additional disk space required) of a mirrored volume. A RAID-5 volume is significantly more efficient than a mirrored volume when larger numbers of disks are used. The space required for storing the parity information is equivalent to 1/Number of disks, so a 10 disk array uses 1/10 of its capacity for parity information.

The utilization increases as the number of disks in the array increases.

## Disadvantages of RAID-5 Volumes

In general, RAID-5 volumes are not well suited for any write intensive workload, since a single write is likely to generate a disk read of the parity and two writes (to update data and to parity).

A RAID-5 volume is not well suited for the following situations:

- Where applications that require high-speed data collection from a process are used. This type of application requires continuous high-speed disk writes, which do not work well with the asymmetrical I/O balance inherent in RAID-5 volumes and the extra I/Os required to write the parity stripe.
- In transaction processing database applications in which records are continually updated, such as in financial applications where balances are frequently updated.
- Where applications that require large sequential data transfers are used. These types of data transfer can prevent effective I/O load balancing.

Neither the system volume nor boot volume can be on a RAID-5 volume. If a disk that is part of a RAID-5 volume fails, read operations for data stripes on that disk are substantially slower than for a single disk. The software has to read all of the other disks in the set to calculate the data.

A RAID-5 volume requires more system memory than a mirrored volume.

**Note** On a computer running Windows 2000 Professional, 32 MB is the minimum amount of RAM; 64 MB is the minimum amount of RAM for a computer that is running Windows 2000 Server. You might want to add memory or use additional services when you configure disks as RAID-5 volumes.

# Configuring and Using Mirrored Volumes and RAID-5 Volumes

This section discusses using features that are available in Windows 2000 Server software to implement mirrored volumes and RAID-5 volumes. It does not discuss configuring or using hardware RAID arrays.

## Creating a Mirrored Volume or RAID-5 Volume

If you are using all SCSI disks, it is a good idea to use the same translation on all SCSI controllers on disks in a mirrored volume or RAID-5 volume. Disable translation when you do not need to multiple-boot with MS–DOS or have a system volume or boot volume larger than 1 gigabyte (GB).

When you create a mirrored or RAID-5 volume, Disk Management allocates unused space from each dynamic disk that you select.

For information about creating mirrored and RAID-5 volumes, see Windows 2000 Server Help.

### Creating a Mirrored Volume

If you need to multiple-boot with MS–DOS, Windows 95, or Windows 98, you cannot mirror your system volume. If you need a boot volume larger than 1 GB, you need to enable translation.

When you want to create a mirrored volume, the original volume that you want to mirror must already exist. You must have an area of unused space on a dynamic disk at least as large as the original volume on another disk connected to the computer.

If the unused space that you select for the volume on the shadow disk is larger than the existing volume, the remaining space is unused and can be used to create other volumes.

### Creating a RAID-5 Volume

To create a RAID-5 volume, you must have at least three unused areas on different dynamic disks connected to the computer. As with a mirrored volume, if the areas of unused space that you select for a RAID-5 volume are not all the same size, the space left over can also be used for other volumes.

When using a RAID-5 volume, performance is better if you put the operating system and page file on the volume and disks used for the RAID-5 volume. You can use different types of controllers for the operating system and the RAID-5 volume when you have them on separate controllers. For more suggestions, see "RAID-5 Volumes" earlier in this chapter.

## Configuring the System Partition on a Mirrored Volume

If your system volume is on a SCSI disk, you need to have the BIOS enabled on the SCSI controller, or the system cannot find the MBR. If you have created a mirror of your system volume by using two SCSI controllers, make sure that you are using the same translation on both controllers.

---

**Important** Note that you cannot create a dynamic volume within Setup. If you must delete a dynamic volume from within Setup, delete all volumes on the disk. This reverts the drive to a basic drive and results in the loss of all data on that drive.

---

Use the translation feature on SCSI controllers to set the translation mode. Because you need to disable the BIOS on the second controller, Windows 2000 always translates the geometry of the shadow volume to 32 sectors per track and 64 heads per cylinder. Therefore, it is important to make sure that the SCSI controller for the original disk is using the same translation. When the BIOS is enabled, most SCSI controllers default to a large disk translation scheme if the disk is larger than 1 GB. The geometry for this translation is 255 heads and 63 sectors. Most SCSI controllers have a method to manually configure the controller to use the 32-sector, 64-head translation. Using 32 sectors and 64 heads on the controller for the original disk ensures that both disks are as close to a true sector-by-sector mirror image as possible. Recovery from failures of the original disk or its controller is *much* easier if the geometry of both disks is the same. The only disadvantage is that older controllers that do not support Interrupt 13 Extensions need the system volume to be within the first 1 GB, or 1024 cylinders, of the disk. Windows 2000 takes advantage of Interrupt 13 Extensions, which makes it possible for the system volume to reside on large volumes that exceed the 1024 cylinder limit

Disabling the controller BIOS is different from turning off translation. Problems occur because you need to have the BIOS enabled on the first controller to start the computer from the hard disk. If you do not turn off translation on both controllers, and the first controller fails, you need to enable the BIOS on the second controller. The translation on the second disk is now different than when you low-level formatted it, and you cannot use the disk.

---

**Note** If you disable the BIOS on both controllers, translation is automatically disabled as well. However, you cannot start up from the hard disk if the BIOS is disabled.

---

If you receive the message "Drive is too small" when you try to reconstruct the mirrored volume after the failure of the original disk, it is likely that the translation is not consistent.

Having two different SCSI controllers complicates restarting the computer in the event of a hardware failure. If you use two different SCSI controllers in your mirrored volume, you need to have a different SCSI device driver for each controller.

---

**Note** The term duplexed mirrored volume means that each disk in the mirrored volume has its own disk controller.

---

You can use Enhanced Integrated Drive Electronics (EIDE) controllers for a mirrored volume of the system volume. EIDE disks usually require motherboard BIOS support for logical block addressing (LBA), and each motherboard usually performs LBA in the same way for all EIDE disks.

---

**Note** If you are going to mirror an EIDE disk with a SCSI disk, do not mirror the system volume. However, you can mirror your boot volume.

---

When the shadow disk or controller fails, recovery is the same as recovering from any mirrored volume failure. Follow the guidelines for identical disks and controllers described earlier in this chapter. For more information about rebuilding the mirrored volume when you have replacement hardware, see "Repair, Recovery, and Restore" in this book and Windows 2000 Server Help.

When the original disk or controller fails, Windows 2000 seamlessly starts using the shadow disk for all accesses. For this reason, a problem can go undetected until you try to restart the computer. In this case, you might need to remove power from the failed disk or remove it from the SCSI bus before attempting to start from the shadow volume using the Windows 2000 startup floppy disk.

**Important** If you use a computer with an Extended Industry Standard Architecture (EISA) volume as the first volume on the primary system disk, you might find that the Windows 2000 startup floppy disk does not start the shadow drive. This is because the Boot.ini is pointing to volume(2) which is not the system volume on the shadow drive. On the shadow drive there is no EISA volume configured as the first volume on the shadow disk. Therefore, you need to change the volume parameter in the Boot.ini to volume(1) as well as changing the disk parameter to point to the shadow drive.

**Important** Try both startup paths before a failure, if possible. Also, you should always check Boot.ini if you are using mirrored volumes.

## Guidelines for Configuring Mass Storage

When choosing a configuration for mass storage, you need to consider variables such as cost, performance, and the time required to completely rebuild a disk. Consider the relative advantages and disadvantages of mirrored volumes, RAID-5 volumes, or a combination of the two techniques.

In general, fault-tolerant configurations are only needed for information that must be readily available in case of hardware failure or unrecoverable disk errors. You do not need to have your page file on a fault-tolerant volume, and it is recommended that you not have the page file on a RAID-5 volume because of potential performance impacts.

You might not need to configure applications on a fault-tolerant volume. If you have applications and other common files on more than one computer running Windows 2000 Server, you can use replication to keep them consistent and to provide the redundancy. If you have applications on a single computer running Windows 2000 Server, you only need to configure them on a fault-tolerant volume if they must always be immediately available. Make sure that you back up the volume every time you install a new application or change default settings for an application.

**Note** Placing application data on a striped volume provides the fastest I/O performance for reading data.

If space is a consideration, format your application volume with NTFS (or convert a FAT volume to an NTFS volume), and use NTFS compression for folders and files on the volume.

## Using Mirrored and RAID-5 Volumes

You need to consider the software and hardware constraints of fault-tolerant storage, as well as the cost and reliability of the system. Each vendor needs to have design guidelines for their system, whether you want a RAID array or want to use the fault-tolerant features provided in Windows 2000 Server.

In certain configurations and situations fault tolerance might not work as you might expect. Check that all of the disk hardware is on the Windows 2000 HCL. If any equipment is not included on the HCL, it might not work well.

If possible, use identical disks. Although this is not a requirement, there are several advantages to using identical hardware for fault-tolerant configurations:

- The performance of the disks are identical. Faster disks do not have to wait for slower disks.
- If you want to configure the entire disk for a mirrored volume or RAID-5 volume, the capacity is the same.
- There are fewer potential problems with configuration and compatibility.

Have a spare disk available and ready for use. When purchasing the disks, it is a good idea to purchase an extra disk for use as a spare. An identical disk guarantees that the spare disk is compatible, does not degrade the system performance, and can be easily installed. If you are using SCSI disks, check the disk documentation to see if you need to manually set the SCSI ID.

Having an available spare or warm spare (a disk that is powered at all times in the configuration and ready to be substituted) allows you to dynamically mirror or replace a member disk of a mirrored or RAID-5 volume in the event of a hardware failure on a spindle.

Have a backup controller available, or configure your disks with duplexed controllers. The system can continue operating when a disk fails, but not when a controller fails. System downtime can be minimized by having a preconfigured controller available. If the replacement controller is not the same as the original, you have to install a new driver, and your configuration becomes more complex.

---

**Note** If you have configured your system volume on a mirrored volume, use the same make and model of controller for your backup, and be sure to use the same translation for both the original and shadow volume.

---

### Configuring Mirrored Volumes

To a large extent, how you configure your mirrored volumes depends on the number of disks and controllers that you want to have on the computer running Windows 2000 Server. Configuring your boot volume on a disk (and controller) that does not contain data sets results in better performance. There are also other considerations that are related to overall reliability.

When you use a mirrored volume for your system or boot partitions, the following issues must be kept in mind:

- You can make the configuration more fault tolerant by putting each disk member of the mirrored volume on separate controllers. This approach allows you to survive controller or disk failures. Putting each disk member of the mirrored volume on a separate channel of a multichannel controller does not make the controller fault tolerant. However, depending on the configuration, it might improve performance.
- Be sure to use the same disk translation for the disks in a mirrored volume of the system volume and to format both disks using the same controller.

---

**Important** Always try starting the computer from both mirrors *before* you have a failure.

---

### Configuring RAID-5 Volumes

When configuring a RAID-5 volume, buy disks on the basis of cost per megabyte and performance, since storage efficiency increases with a larger number of disks. If you use high performance SCSI controllers, an array of six 9-GB disks provides more efficient storage than nine 6-GB disks. Smaller disks can also be rebuilt faster.

Keep the RAID-5 volume on a different controller and disk than your system and boot volume. Using separate controllers improves performance and can accelerate recovery from hardware failures. Remember that you cannot configure your system volume or your boot volume on a RAID-5 volume.

If the computer itself or the disk containing the boot volume fails (and the boot volume is not mirrored), it might be faster to move the RAID-5 volume to a different computer.

## Fault-Tolerant Hardware Versus Software

The following are points to consider when deciding between a fault-tolerant hardware or software solution:

- Fault-tolerant software is available only on Windows 2000 Server.
- Hardware fault tolerance is faster.

- Software fault tolerance is less expensive.
- In a hardware fault tolerance implementation, some vendors support hot-swappable disks when there is a disk failure.

Regardless of whether you implement fault tolerance by using hardware or software, implementing fault tolerance does not reduce the need for backups.

## Testing a Fault-Tolerant System

If the computer has software fault-tolerant volumes, test your configuration by simulating the failure and replacement of a disk.

Even though fault-tolerant volumes continue to work when one disk has failed, there is no fault tolerance until you install a replacement disk. A second disk failure during this interval will result in loss of data, because data redundancy was lost when the first failure occurred. If a backup disk is not readily available, significant business costs can result from the downtime.

If you have hardware RAID arrays, documentation provided by the vendor needs to describe how to recover from disk or controller failures. Make sure to test that their procedures work for your installation.

**Important** Fault tolerance is *not* an alternative to performing regular backups.

# Avoiding Single Points of Failure

Although it is important to be prepared for problems, you can take steps to protect against certain failures, such as disk failures, component problems, network problems, and power failures. Hardware and software configurations can be implemented to help reduce the likelihood of problems that result in costly downtime and recovery processes.

Computers running Windows 2000 Server have fault tolerance features built into the operating system. Typically, fault tolerance applies to disk subsystems, but it can also apply to other parts of a system or an entire system. For example, a fully fault-tolerant computer system uses redundant disk controllers and a UPS as well as fault-tolerant disk subsystems.

Although data is always available and current in a fault-tolerant disk configuration, backups are always needed to protect data on a disk subsystem from failures caused by unpredictable occurrences such as user error, sabotage, or software malfunction, and catastrophes such as fires, earthquakes, or floods. Keeping replacement disks and controllers available on-site is cost-effective.

Disk fault tolerance is not an alternative to a backup strategy. Your backup strategy should include off-site storage.

Consider providing UPS protection for individual computers and the network itself, including hubs, bridges, and routers. Windows 2000 has UPS support on individual computers which typically provides power long enough for Windows 2000 to complete an orderly shutdown when power fails. If your company has a history of frequent or prolonged power outages, investigate alternative power sources besides the local power company for critical computers. It is important to remember that even a UPS for every computer on the network does not necessarily prevent data loss or corruption due to power fluctuations. The network is itself an electrical system. Intermediary devices such as routers, bridges, and hubs, require the same UPS protection to prevent loss of network functionality.

# Avoiding Computer Component Failure

Typically, server failure is the most costly for a business, whether it is a file server, a print server, or an applications server. You need to consider the effects of individual components when deciding how to configure company computers and when to run diagnostic tests.

## Motherboard and Central Processing Unit

Motherboards consist of electronics that can and do fail, although the motherboard and the CPU are generally reliable computer components. There is little that you can do to avoid a motherboard failure or CPU fault, except to run regular system checks to ensure that the components are functioning correctly. Some systems include built-in diagnostic tools that operate with Windows 2000.

## RAM

The three major types of RAM that deal with error detection and correction are parity RAM, error-correction coding (ECC) RAM, and nonparity RAM.

**Parity RAM** Parity RAM contains an extra bit that indicates if each byte in the RAM is faulty. When parity RAM detects a parity difference, it signals the CPU through a nonmaskable interrupt (NMI). Depending on where and when detection happens, Windows 2000 determines if this is an I/O board parity error, memory bus error, or some other kind of parity error. Windows 2000 can also report I/O channel parity errors from cards in slots. This generates an error message in these cases, and sometimes the computer stops.

**Error-correction coding RAM** High-end systems often use ECC RAM, which can detect a two-bit failure and correct a single-bit failure in the system memory. Windows 2000 continues to run in spite of a single-bit failure. Depending on the hardware design, there might or might not be a report of this corrective action.

**Nonparity RAM** If you use nonparity RAM, Windows 2000 has no way to detect memory problems, and your computer might crash randomly. Nonparity RAM costs less than parity RAM, and parity RAM is not available for all computers. If you do not have parity RAM in your computers, ask your vendor if it can be installed or is supported by the computer.

Some vendors supply products that you can use to check the RAM in a computer.

## Video Cards

Video cards drive the screen and render images for display. They rarely cause computer failures, but might cause the computer to behave erratically. More often, video cards cause screen painting problems, application page faults, and the like. Such problems are typically not critical enough to require you to shut down the computer. To minimize video problems, be sure that your computer is running the most recent release of a supported video driver.

## Disks and Disk Controllers

You have many choices for your disk configuration, including fault tolerant configurations. EIDE and SCSI technologies each offer different benefits for fault tolerance and recovery. The MTBF gives you a measure of expected disk and controller reliability.

Be sure to run disk and controller diagnostics during every preventive maintenance check. Diagnostics are typically available from your hardware vendor. Windows 2000 automatically runs Chkdsk every time you start up the computer, and you can run a surface scan of the disks by specifying **chkdsk /r**. Chkdsk is also available from the Recovery Console.

## Network Adapters

Asynchronous Transfer Mode (ATM) and other network adapters can have dual channel connections. If one channel fails, the other is automatically used.

Ethernet and Token Ring network adapters do not have dual channel capability. If the manufacturer provides a diagnostic program, it is recommended that you run it on the network adapterss during scheduled preventive maintenance or downtimes.

You can evaluate network segments with network packet trace programs, called sniffers. Network Monitor can check for the following problems:

- Invalid Cyclic Redundancy Checks (CRCs).
- Corrupted packets.
- Bandwidth saturation caused by a broadcast intensive network adapter.

# Using an Uninterruptible Power Supply

An *uninterruptible power supply (UPS)* provides constant power to a computer system when a power fluctuation or power loss occurs. Built-in electronics constantly monitor line voltages: if the line voltage fluctuates above or below predefined limits, or fails entirely, the UPS supplies power to the computer system from built-in batteries. The UPS converts the direct current (DC) battery voltage into the alternating current (AC) voltage required by the computer system. The change to batteries must take place very rapidly to prevent data loss.

Most UPS devices are one of the following types:

- Online UPS. An online UPS is connected between the main power and the computer that constantly supplies power to your computer. The main power continuously charges the batteries which supply the power to the computer. Connecting it to the main power keeps its battery charged. This method provides power conditioning, which means that it removes spikes, surges, sags, and noise.
- Standby UPS. A device configured to provide either the main power or its own power source and to switch from one to the other as necessary. When main power is available, the UPS device connects the main power directly to the computer and monitors the main power voltage level. When the main power fails or the voltage falls below an acceptable level, the UPS device switches to its own power.

Use only UPS hardware that is included on the Windows 2000 HCL. Other options to consider include:

- Whether to use a separate UPS for each computer or have larger capacity, centralized UPSs that protect multiple computers.
- What type of UPS you want to use.
- How big the UPS needs to be to protect your computer systems.
- How long the UPS needs to run before automatically shutting down.
- What other features your UPS needs to have, such as:
  - Continuous conditioning of the incoming power to provide clean, steady power.
  - Software that produces statistics or logs UPS information to the event log.
  - Software for testing the integrity and reliability of the UPS battery.

For robust resistance to power failures, use UPS hardware connected to the computer and software that handles power failures, including shutting down the system before the UPS batteries are depleted. Without such software, human intervention is needed to shut down the system.

Windows 2000 has built-in UPS functionality that takes advantage of the features included with many UPS systems, such as ensuring the integrity of data on the system by providing for an orderly shutdown of the computer system if a power failure lasts long enough to deplete the UPS batteries. In addition, users connected to a computer running Windows 2000 Server can be notified that a shutdown will occur, and new users are prevented from connecting. Finally, damage to the hardware from a sudden, uncontrolled shutdown can be prevented.

Some vendors also provide a user interface for configuring the UPS, which you can use instead of the one provided in Windows 2000.

To fully protect your network, you also need to install a UPS on network devices such as routers, hubs, and bridges. For the best protection, install UPS systems on the cables that connect your computer to your modem, telephone, printer, and network equipment. Most UPS systems have built-in surge protection.

In Control Panel, click **Power Options** to configure the Windows 2000 UPS service.

▶ **To configure the UPS service in Windows 2000**

1. In Control Panel, select the **Power Options** option.
2. Select the **UPS** tab, and then click the **Select** button.
3. From the **UPS Select** dialog box, select the manufacturer and model of the UPS unit you are using.
4. In the **On port** field, select the port to which the UPS is connected, and then click **Finish**.
5. On the UPS page of the **Power Options Properties** dialog box, click the **Configure** button.
6. From the **UPS Configuration** dialog box, select the configuration options that best suit your needs.
7. Click **OK**.
8. On the **UPS** page of the **Power Options Properties** dialog box, click **OK** or **Apply** to commit the new settings.

The remaining information depends upon your specific UPS and its use.

When selecting the signals to use in the **UPS Configuration** group box, the interface voltages indicate the active state for the signal. For example, if you select a negative interface voltage for the power failure signal, the normally positive signal becomes negative when a power failure occurs.

Use the following procedure to configure the Windows 2000 UPS service to run a command file when a power failure occurs.

▶ **To configure the UPS service in Windows 2000 to run a command file when a power failure occurs**

1. In the **UPS Configuration** dialog box, select the **When the alarm occurs, run this program** check box, and then click **Configure**.
2. In the **UPS System Shutdown Program** dialog box, select the **Task** tab.
3. Type the name of the file in the **Run** field.
4. If you want the scheduled task to run at a specified time, select the **Enabled (scheduled task runs at specified time)** check box and continue this procedure.
5. Select the **Schedule** tab, and then select the frequency and start time of the task.
6. For further schedule settings, click the **Settings** tab.
7. Choose from options in the **Scheduled Task Completed**, **Idle Time** and **Power Management** areas of the **Settings** tab, then click **OK**.
8. In the **Set Account Information** dialog box, type the **Run as** and password information, and then click **OK**.

---

**Note** The command file must reside in %SystemRoot%\System32 and have a file name extension such as .exe, .com, .bat, or .cmd.

---

# Avoiding Disruptions to Network Connections

If users cannot connect to the computers in a configuration that is running Windows 2000 Server, that configuration is not fault tolerant. You need to consider what can fail in the connections between computers as well as within an individual computer.

## Network Cabling

If your company is growing, the connections between the computers that make up your configuration could become saturated with network traffic. It is recommended that you evaluate network traffic regularly to determine your need to upgrade by adding equipment. You can also use newer technologies, such as ATM, Fast Ethernet 100 BaseT, or GB Ethernet.

## Intermediary Devices

Devices that connect different segments of your network, such as routers, bridges, hubs, and switches can also cause bottlenecks and failures. Intermediary devices need to be protected with a UPS.

Determine the level of vendor support available for each device, as well as the standards supported by each device, and whether the vendor has a migration path for new standards. You also need to find out if there are frequent software changes.

### External Network Connections

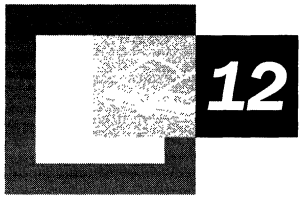
If you lease an X.25 connection, an Integrated Services Digital Network (ISDN), Asymmetric Digital Subscriber Line (ADSL), or T1 line to connect to another building, branch, or subsidiary of your company, verify that your line vendor has adequate recovery procedures in place.

## Additional Resources

- For more information about RAID volumes, see Windows 2000 Server Help.
- For more information about mirrored and RAID-5 volumes, see Windows 2000 Server Help.

CHAPTER 12

# Backup




Backing up system state data, files, and other data on your system is an important aspect of planning a reliable configuration. If you do not back up your data, you might not be able to recover important information or settings when problems occur. Microsoft® Windows® 2000 provides a backup utility that you can use to make sure that up-to-date copies of your data can be readily restored.

### In This Chapter

Overview of Backups 495
Establishing a Backup Plan 501
Backing Up System State Data 511
Using Backup 515
Backing Up Windows 2000 Services and NTFS Features 520

### Related Information in the Resource Kit

- For more information about planning considerations, see "Planning a Reliable Configuration" in this book.
- For more information about restoring from backups, see "Repair, Recovery, and Restore" in this book.
- For more information about data storage options, see "Data Storage and Management" in this book.

# Overview of Backups

Regular backup of servers and local hard disks prevents data loss and damage caused by disk drive failures, power outages, virus infection, and many other possible computer problems. Backup operations based on careful planning and reliable equipment make file recovery easier and less time consuming.

## Backup Types

Backup can perform several types of backups:

- A normal backup copies all selected files and marks each as having been backed up. With normal backups, you need only the most recent copy of the backup file to restore all of the files.
- An incremental backup backs up only those files created or changed since the last normal or incremental backup. It marks files as having been backed up. If you use a combination of normal and incremental backups, you need the last normal backup set as well as all the incremental backup sets to restore your data.
- A differential backup copies files created or changed since the last normal or incremental backup. It does not mark files as having been backed up. If you are doing normal and differential backups, you must have the last normal and last differential backup sets to restore.
- A copy backup copies all selected files but does not mark each file as having been backed up. Copying is useful to back up files between normal and incremental backups, because it does not affect other backup operations.
- A daily backup copies all selected files that have been modified on the day that the daily backup is performed. The backed up files are not marked as having been backed up.

**Note** Perform regular backups when the fewest people are using the network. If many files are in use, the backup might not accurately reflect your network.

## Storage Devices and Media

Windows 2000 can back up files to a variety of storage devices, including a tape drive. Data can be backed up to a logical drive, removable disk, network share, or a library of disks or tapes organized into a media pool and controlled by a robotic changer. If you do not have a separate storage device, you can back up to another hard disk or to a floppy disk.

## Storage Devices

Storage technology changes rapidly, so it is important to research the merits of various media before you make a purchase. Consider drive and media costs, as well as reliability and capacity, when selecting a storage device. Ideally, a storage device has more than enough capacity to back up your largest server and can also detect and correct errors during backup-and-restore operations. For information about supported storage devices, see the Windows 2000 Hardware Compatibility List (HCL) link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

**Tip** The best way to make sure that your storage devices and media are working correctly is to verify your backups by performing test restores.

## Media Types

The most common type of medium is magnetic tape. The primary tape drives used for backup include a quarter-inch cartridge (QIC), digital audio tape (DAT), 8mm cassette, and digital linear tape (DLT). High-capacity, high-performance tape drives typically use SCSI controllers. Other types of media include magnetic disks, optical disks and CD-ROMs (CD-R and CD-RW).

# Backup Strategies

A good backup strategy is the best defense against data loss. The following are three common backup strategies with questions to consider when determining what is appropriate for your computer system.

**Network backup or server only** Do you plan to back up your entire network, or do you have storage devices attached to certain servers where users copy their important files?

**Individual or local computer backup** Does each computer need a storage device? Is each user responsible for backing up his or her data?

**Server and computer backup** Does each department have a storage device and a designated user to back up all data for that department?

You can physically connect your storage device to a local computer or to a server. Table 12.1 lists some advantages and disadvantages of each configuration.

**Table 12.1 Comparison of Server-Only and Local Computer Backups**

| Backup Type | Advantages | Disadvantages |
|---|---|---|
| Server-only | Fewer storage devices are needed. | Registries and event logs of remote computers are not backed up. |
| | There is less media to manage because shared media stores multiple backups. | Backups and restorations are slower, due to network throughput. |
| | If each server has more than one client computer, server-only backup costs less than backing up individual computers. | Backups and restorations require greater planning and preparation. They must be scheduled when network traffic is low or when critical information can be backed up as quickly as possible. |
| Local computer | Fewer network resources are committed to lengthy backup procedures. | Using more storage devices costs more. |
| | File recovery is quicker. | |

Backups that include computers running both Microsoft® Windows® 2000 Server and Microsoft® Windows® 2000 Professional combine the advantages and disadvantages of server-only and local computer backups. Additional considerations include the following:

- If you run backups from a server, you can back up and restore that server quickly. However, if your network has more than one server, the speed advantage is lost.
- Servers typically operate 24 hours a day, so it is possible to run backups during less busy hours. However, if a problem with backup hardware occurs, you might have to turn off the server.
- As a rule, if data is primarily on one server, it is better to run backups from that server.

**Note** Whether you run remote backups from a local computer or a server, it makes sense to place the storage device on the portion of your network with the greatest bandwidth or highest transmission frequency. You might also consider keeping the storage device in a secure room.

Figure 12.1 illustrates local computer and server-only backup options for a tape drive.

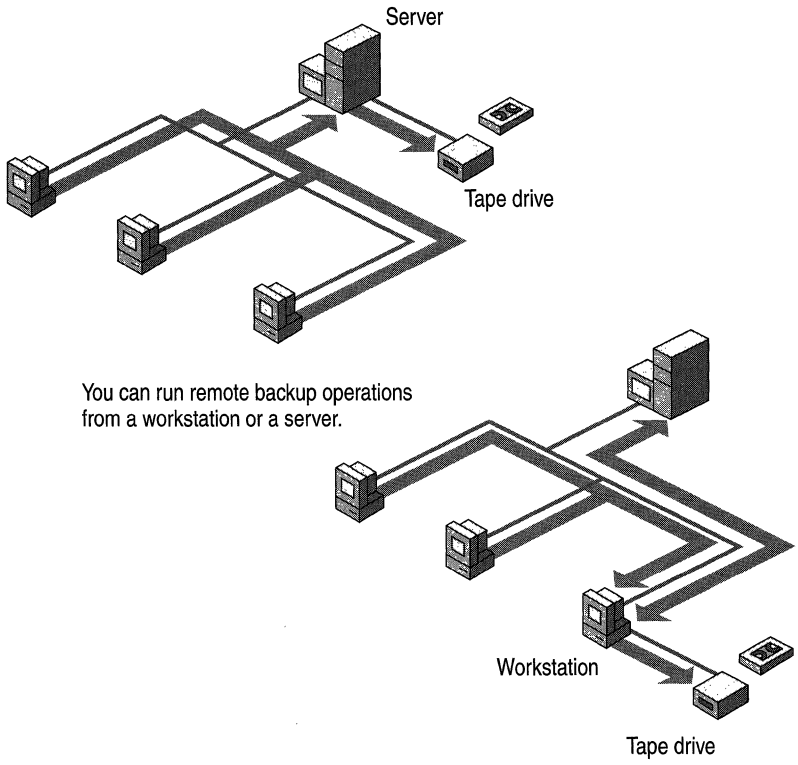



**Figure 12.1 Server-Only and Local Computer Backup Configurations**

**Caution** Backups protect data against viruses. Because some viruses take weeks to appear, it is recommended that you keep normal backup tapes for at least a month to make sure that you can restore a system to its uninfected status.

# Security Considerations

There are several steps that you can take to enhance the security and operation of your backup-and-restore operations. You should also take steps to secure your backup media.

When you form a backup plan, consider implementing the following methods:

- Secure both the storage device and the backup media. Data from stolen media can be accessed and restored to another server.
- Back up an entire volume in case of a disk failure. It is more efficient to restore the entire volume in one operation.

- Always back up the system state data on a domain controller to prevent the loss of user accounts and security information.
- Keep three current copies of backup media. Store at least one copy off-site in a properly controlled environment.

## Backup and Restore Rights

Backup and restore rights are independent of each other. However, you can grant both rights together.

**Important** You must have administrative privileges to restore the system, but only restore privileges are required to restore files.

If you are the system administrator, extend backup and restore rights only to persons responsible for backing up your network. In a minimum-security or medium-security network, grant one user backup rights and another user restore rights. Train personnel with restore rights to perform all restore tasks in the event that you are unavailable. In a high-security network, only you should restore files.

**Caution** A person who does not have permission to write to a file might have permission to restore the file. During restoration, such permission conflicts are ignored, and the existing file can be overwritten.

## Granting Rights at Large Sites

Large sites might have two groups of backup operators: one with backup rights only; and the other with restore rights. It is recommended that you implement the following guidelines:

- Grant backup-only rights by creating a local group named Backup Operators and give members **Backup files and directories** user rights. Then, create a global group named Backup Only and add it to the local group.
- Grant restore-only rights by creating a local group named Restore Operators and give members **Restore files and directories** user rights. Then, create a global group named Restore Only and add it to the local group.

## File Permissions

In Windows 2000, access to NTFS files is limited by NTFS file permissions, share permissions, and file attributes. You cannot back up or restore NTFS files to which you do not have access right unless:

- You belong to the permissions group of administrators, backup operators, or restore operators.

- You have been granted user rights to **Backup files and directories** (if you are backing up) and **Restore files and directories** (if you are restoring).

**Note** Neither the FAT16 nor the FAT32 file systems provide file permissions.

## Storing Backup Media

Some kinds of information need to be stored near your computers to be readily available; store some data off-site to be available in the event of a disaster or for long-term storage.

**Caution** Backup media lasts longer in cool, humidity-controlled locations. Your storage area should also be free of magnetic fields, such as those near the backs of computer terminals and analog telephone equipment.

### On-Site Storage

Some data needs to be stored on-site and readily available in the event of a failure.

**Daily backups, whether full or incremental** Store media in a fireproof safe or cabinet. Secure storage protects against natural disaster, theft, and sabotage.

**Copies of the media** If more than one copy of a software program is purchased, store one off-site if possible. If you have only one copy, you can back it up to media and label it as a backup of the application. If you need to reinstall software, you can restore from media to a computer running Windows 2000. You can then run the application setup program over the network and delete the software from the server.

### Off-Site Storage

For highly confidential data that must be stored off-site, consider assistance from a company that specializes in secure data storage. Such companies lease space in underground vaults that remain impervious to most threats. If the cost or logistics of such protection is too great, use an alternative solution, such as a safe-deposit box or an off-site fireproof safe designed to protect magnetic media.

It is recommended that you store the following items off-site:

- A full backup of the entire system, best performed weekly.
- Original software installed on computers (only copies kept on site).
- Documents required for processing an insurance claim, such as purchase orders or receipts for computer hardware and software. (Pertinent information is available from your business insurance agent.)

- Information required to get network hardware reinstalled or reconfigured.
- Information required to reconfigure your storage subsystem.

**Tip** Make sure that your off-site storage location is bonded.

# Establishing a Backup Plan

When developing your backup plan, keep the following in mind:

- Be sure to have spare hardware in case of a failure in your backup device.
- Test backed up data regularly to verify the reliability of your backup procedures and equipment.
- Include stress testing of backup hardware (storage drives, optical drives, and controllers) and software (backup program and device drivers).

## Backup Scenarios

There is a wide range of system configurations that can affect your backup strategies. At one extreme is a small, simple network with a computer that must be backed up and also be available 24 hours a day, seven days a week; at the other is a computer with a large amount of data that needs to be backed up. This section discusses backup strategies for these two situations.

**Caution** Backup does not back up files on computers running Microsoft® MS-DOS®. Consider reserving some space on the server so that users of MS-DOS and Microsoft® Windows version 3.1 can copy important files. Those files can be backed up during regular server backups.

### Small and Medium LANs

The following scenario illustrates a possible approach for backing up a small network comprised of a computer that is running Windows 2000 Server and 20 client computers:

**The situation** The clients are a mix of computers running MS-DOS, Microsoft® Windows® for Workgroups, Microsoft® Windows® 95, Microsoft® Windows® 98, Microsoft® Windows NT®, and Windows 2000. You plan to purchase one tape drive and a controller card and to use Backup.

**An approach** You can work out a backup solution by following these four steps:

1. Research and select a tape drive for its reliability, speed, capacity, cost, and compatibility with Windows 2000. The drive should support tape cartridges with more than enough space to back up your entire server.
2. Install the tape controller card in the server. If you are using SCSI, put the tape drive on its own controller.
3. Connect your tape drive to the server so that you can back up the server system state data and so that server backups are most efficient. If you are using a SCSI tape driver, turn on the tape drive before turning on the server so that the SCSI tape driver can be loaded properly. From the server, you can back up user files on remote computers that are running Windows for Workgroups, Windows 95, Windows 98, Windows NT, and Windows 2000.
4. Establish a tape rotation schedule for backing up computers running Windows for Workgroups, Windows 95, Windows 98, Windows NT, and Windows 2000 Professional. To conserve tapes, back up clients less frequently than the server and encourage users to copy critical files to the server at the end of the day. Figure 12.2 shows one possible tape rotation schedule.

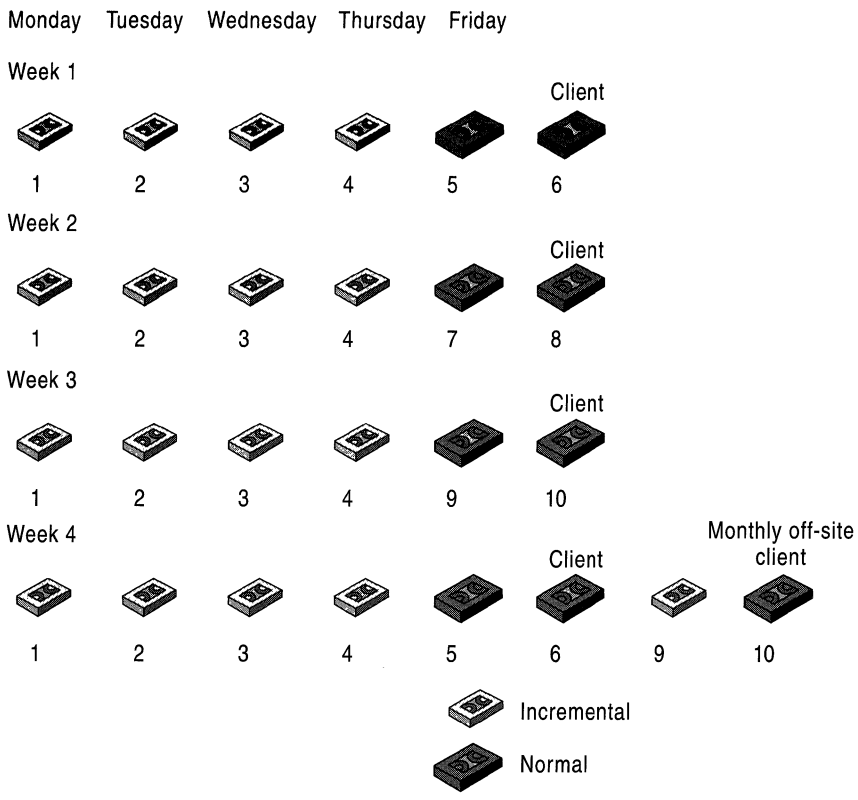



**Figure 12.2 Possible Rotation Schedule for Backup**

## Methods for 24-Hour Operations or Very Large Backups

If you need to back up a lot of data, such as a Microsoft® SQL Server™ database or a number of large graphics files, it can take a long time. When the time available for backups is short, approaches that work for small and medium-sized local area networks (LANs) become inadequate. It is helpful if you use a backup utility that can back up while the application remains available to clients.

You can configure the disks that require daily backup separately from the Windows 2000 system disks. For critical data, use host-based or hardware-based redundant array of independent disks (RAID). For very critical data, you can even implement a software mirror of two separate hardware-controlled RAID arrays. With this configuration, if either a disk or an entire array fails, operations continue. If a component such as a network adapter, video device, IDE adapter, or power supply fails, it can easily be replaced. It is recommended that you have a spare computer with Windows 2000 Server installed as a backup for data. For more information about RAID arrays, see "Planning a Reliable Configuration" in this book.

---

**Important** Hardware reliability does not replace the need for regular backups. Backing up data protects you against software problems as well as human error.

---

In the descriptions of the following methods, the computer that contains the data to be backed up is called Data. The computer that runs the backup is called Target. Unlike the Methods One and Two, Method Three allows users to continue to access Data while it is being backed up.

## Method One

Back up Data locally to disk. Use a networked backup or Xcopy to move the resulting backup file to Target. Make sure to regularly run a backup verification pass comparing the data on Data and Target. Do not use Compare when you are using the Active Directory™ directory service because comparing data takes about twice as long as copying it. Typically, Ethernet or Token Ring transfer speeds are 1 megabyte (MB) per second if the network is not busy. You can use this transfer rate and the total amount of data being transferred to estimate the transfer time. You might need to use a faster network connection or a different backup method.

## Method Two

Back up Data over the network to Target. Copy the data that you want to back up to another disk or disks on Data. Bring Data online, and copy the data using a backup device connected to Data. You can also back up Data over the network to Target. Whether to perform the backup from Data or from Target depends on the following factors:

- The availability of a target computer.
- Any backup policies requiring that backups be performed on designated computers.
- The time and cost of performing the backup from Data versus transferring the files to Target.

### Method Three

Use hardware mirroring and third-party utilities that can mirror data across the network to another computer running Windows 2000 while the files are being used on Data. For more information about two-site backups, refer to the documentation provided by your hardware or software vendor. Mirroring data is best when you cannot risk losing data because a disk or RAID array fails or when you must protect against loss of an entire site. While there might be some downtime while moving data from the mirror computer to Data after a failure on Data is fixed, it is preferable to losing all data created or changed since the last backup.

**Important** Backup file servers often. If the data is critical, it must be mandatory for users to close their files at the end of the day.

## Establishing a Backup Schedule

The frequency with which you create backups depends on how often your data changes and how valuable your data is. You can perform a weekly backup, a monthly backup, and an archive backup. The archive backup can be a simple copy.

### Twelve-Week Schedule

The 12-week schedule uses a different backup tape each day for two weeks, with the first tape being used again at the beginning of every third week. Incremental backups are performed Monday through Thursday, and normal backups are performed Friday, with the most recent normal backup stored on-site, as illustrated in Figure 12.3. The normal backup of the preceding week is stored off-site. At the end of 12 weeks, the cycle starts again with a new set of tapes.

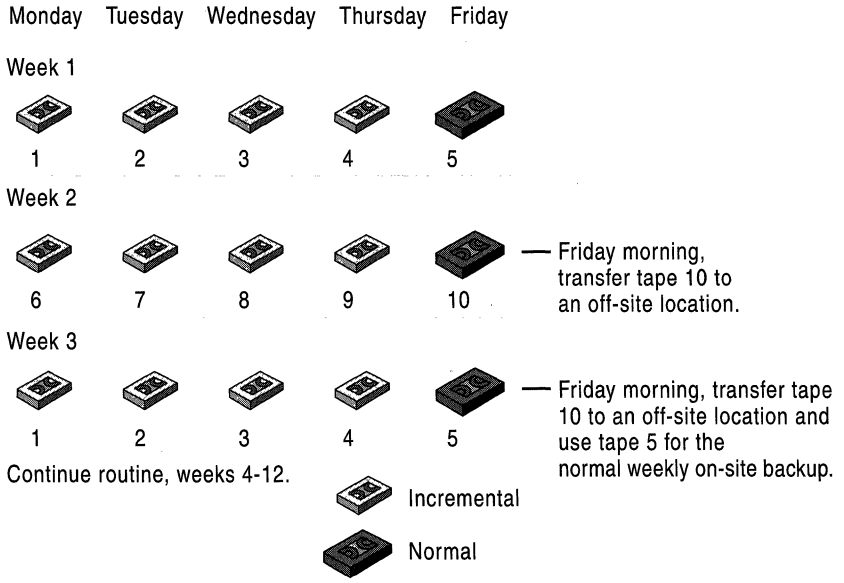



**Figure 12.3 Backup Rotation for a 12-Week Schedule**

## Twelve-Month Schedule

One popular tape-alternation schedule uses 19 tapes over the course of one year. Four tapes are used Monday through Thursday for incremental backups, and three tapes are used for the normal backups performed each Friday. Figure 12.4 shows one month of the 12-month schedule. The remaining 12 tapes are used for monthly normal backups that are stored off-site.

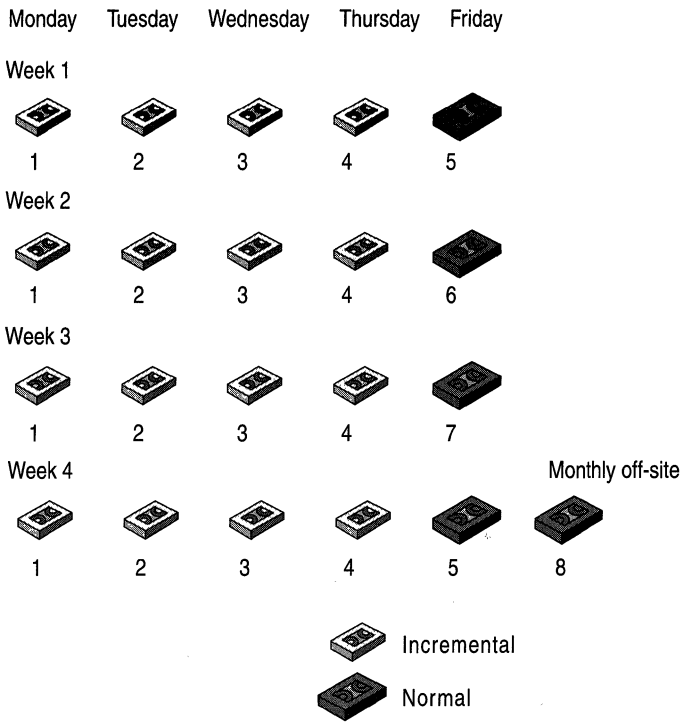



**Figure 12.4 Backup Rotation for One Month in a 12-Month Schedule**

**Note** You can lower backup costs by alternating backup media. The life cycle of a tape depends on the manufacturer and on storage conditions.

# Developing Backup and Restore Procedures

Backup specifics and procedures vary according to the needs of a company. After you develop your procedures based on the considerations discussed in this section, it is important to test, document, and verify them. Periodically reviewing your backup-and-restore process is a key part of ensuring data security.

## Considerations for Developing Procedures

This section offers a series of checklists—suggestions and questions—to serve as flexible guidelines when you develop a comprehensive backup-and-restore process for your company.

## Delegation of Tasks

It is critical that reliable personnel perform your backup and restore operations. Consider the following questions when deciding how to delegate these tasks:

- Who makes the policy that determines what files and computers are backed up, and how is the policy made known?
- Who is responsible for performing backups?
- If backups occur automatically, who handles interruptions such as error messages?
- Who does the backup when the assigned backup operator is unavailable?
- To whom is the success or failure of a backup reported? Who notifies the users if a backup fails?

## Time-Sensitive Backup Questions

In addition to determining when and how often backups take place, it is important to know how long it takes to retrieve backup media and perform a restore. To determine this, ask the following questions:

- Do all backups, both full and partial, occur outside of regular business hours?
- Should backups occur immediately after or before regular business hours?
- How often do you perform full and incremental backups?
- How long does it take to retrieve backups or copies from a local or remote storage area? Can the remote copies be obtained at any time or only during business hours?
- How long does it take to perform a full restore if the computer fails?

## In the Event of a Backup Problem

Make sure to take certain issues into consideration before a backup problem occurs:

- Determine who is notified if a problem or backup failure occurs, and what process they follow.
- If backup fails because of hardware problems, is standby hardware on-site or available on loan from your vendor? Determine how long it takes to replace failed hardware.
- Determine the availability of hardware and software technical support.
- Does your technical support staff have configuration information about computers running Windows 2000? If not, you need to determine how to make the information available in the event of a problem.
- How does support from hardware or software vendors affect how long fixes take?

- If trained personnel monitor overnight backups, are they also scheduled to work the next day? Who covers their other duties while they troubleshoot backup problems or restore data?

## Security Considerations

The security of your backup operations, as well as the security of the storage location, is of paramount importance. Take the following questions into consideration when planning for the security of company data:

- Where are the backup tapes stored?
- What has been done to make the backup location secure in the event of fire, flood, theft, or another disaster?
- How is the backup location monitored?
- Are the tapes that are stored on-site always accessible to the people who need them?
- If there are copies of backup media, where are they stored?
- Is the backup location bonded?

## Policy Considerations

Developing a backup-and-restore process and deciding what to back up requires that you either set or comply with company policy. Keep the following issues in mind when determining your backup plans:

- What is the policy for backup, and how is your plan in compliance?
- Are all modified files to be backed up, or does company policy specify only critical files or the files of certain users, groups, departments, or divisions?
- Are any disks or volumes on the computer not to be backed up?
- Are users responsible for backing up their own client systems?
- Is there a charge-back system for the amount of storage used?
- How is the backup process validated?

## Technical Considerations

You also need to determine how your backups are performed. For instance, determine the following:

- Does the system have to meet certain conditions before the backup starts?
- How is the backup started—from the command line, from an icon, or by batch?
- Are logs created and are they correct?
- If the path is long, the file name odd, the file size very large, or the number of files large, does the backup still work? Can you restore files that have these characteristics?

- Is the backup done to a local tape drive, remotely over the LAN, or remotely over the wide area network (WAN)?
- Does the tape verify that the backed-up data is correct?
- How are the connections between a data source and the storage device verified before the backup begins?
- Are computers equipped for power outages if operators are not present when backups take place?
- Do backups occur on schedule?
- What is the process in place for dealing with unforeseen occurrences during a backup or restore?

## Testing Backup-and-Restore Procedures

Complete verification of the entire backup-and-restore process is critical. Develop backup-and-restore strategies with appropriate resources and personnel, and then test them. Testing backup strategies also demonstrates how much time is required to restore data. A good plan ensures fast recovery of lost data.

Try performing a trial restoration periodically to verify that your files are properly backed up. A trial restoration can uncover hardware problems that do not show up with software verifications.

After a backup strategy has been designed, test it thoroughly with as many simulated failures as possible. For example, if you use disk mirroring, simulate a disk failure by removing or powering down one of the mirrors and ensure that remaining mirror continues to operate without interruption. Again, while RAID is effective, it does not eliminate the need for backups. A data recovery plan based on disk mirroring alone offers no protection if a computer is stolen.

The following questions can help you assess your verification strategy.

- Has every option that you expect to use been tested?
- Do automated backup instructions work?
- Does the backup-and-restore process work properly?
- After you make changes to the operating system (such as installing a service pack), or the backup program, does the backup-and-restore process work properly?
- After you make hardware changes (such as installing a new controller or tape drive, or changing the BIOS on the motherboard) does the backup-and-restore process work properly?
- When you change the hardware or software involved in a backup, how do you verify that you can use the old tapes?

## Documenting Backup-and-Restore Procedures

Keeping accurate backup records is essential to finding missing information quickly, particularly if you have accumulated a large number of high-volume media. Thorough records include media labels, catalogs, and online log files and log books.

**Media labels** Labels should contain a date, the type of backup (normal, incremental, or differential), and a list of contents. If you are restoring from differential or incremental backups, you need to locate the last normal backup and either the last differential backup or all incremental backups created since the last normal backup. Alternately, you can label media sequentially and keep a log book of media content.

**Catalogs** Most backup software includes a mechanism for cataloging backup files. Backup stores backup catalogs on the backup media and temporarily loads them into memory. Catalogs are created for each backup set, or collection of backed up files from one drive.

**Log files** Log files include the names of all backed up and restored files and directories. A log file is useful when restoring data because you can print or read it from any text editor. Keeping printed logs in a notebook makes it easier to locate specific files. For example, if the tape containing the catalog of the backup set is corrupt, you can use the printed logs to locate a file.

## Conducting Verify Operations

A verify operation compares the files on disk to the files on backup media. It occurs after all files are backed up or restored, and it takes about as long as the backup procedure. It is recommended that you perform a verify operation after every backup, especially if you back up to a set of media for long-term storage. A verify operation is also recommended after file recovery.

---

**Note** If a verify operation fails for a given file, check the date that the file was last modified. If the file changes between a backup and a verify operation, the verify operation fails.

---

# Backing Up System State Data

System state data is comprised of the following files:

- Boot files, including the system files, and all files protected by Windows File Protection (WFP).
- Active Directory (on a domain controller only).
- Sysvol (on a domain controller only).

- Certificate Services (on certification authority only).
- Cluster database (on a cluster node only).
- The registry.
- Performance counter configuration information.
- Component Services Class registration database.

The system state data can be backed up in any order. Restoration of the system state replaces boot files first and commits the system hive of the registry as a final step in the process.

System state backup and restore operations include all system state data: you cannot choose to backup or restore individual components due to dependencies among the system state components. However, you can restore system state data to an alternate location in which only the registry files, Sysvol directory files, and system boot files are restored. The Active Directory database, Certificate Services database, and Component Services Class Registration database are not restored to the alternate location.

Although you cannot change which components of the system state are backed up, you can back up all protected system files with the system state data by setting advanced backup options.

### ▶ To set advanced backup options

1. In the **Backup** dialog box, click the **Backup** tab, and then select the files and folders that you want to back up.
2. Click **Start Backup**.
3. In the **Backup Job Information** dialog box, click **Advanced**.
4. Set the advanced backup options that you want, and then click **OK**.

The advanced backup options are described in Table 12.2.

**Table 12.2 Advanced Backup Options**

| Option | Description |
|---|---|
| Back up data that is in Remote Storage | Backs up data that has been designated for Remote Storage. You can restore Remote Storage data only to an NTFS volume that is used with Windows 2000. |
| Verify data after backup | Verifies that the backed up data is exactly the same as the original data. This could substantially increase the time it takes to perform a backup. |

*(continued)*

**Table 12.2 Advanced Backup Options *(continued)***

| Option | Description |
|---|---|
| If possible, compress the backup data to save space | Compresses the data you are backing up so you can save more data on a tape. If this option is disabled, you do not have a tape drive on your computer or your tape drive cannot compress data. |
| Automatically back up system protected files with the system state | Backs up all of the system files that are in your %SystemRoot% directory in addition to the boot files that are included with the system state data. Substantially increase the size of your backup job. |

**Note** If you have more than one domain controller in your organization, and Active Directory is replicated to any of these other servers, you might have to authoritatively restore any Active Directory data that you want to restore. For more information about authoritative restores, see the *Microsoft® Windows® 2000 Server Resource Kit Distributed Systems Guide*.

### ▶ To back up system state data

1. Start Backup.
2. Click the **Backup** tab, and then select the **System State** check box.

   This backs up the system state data along with any other data that you have selected for the current backup operation.

Keep the following in mind when you are backing up system state data:

- You must be an administrator or a backup operator to back up files and folders.
- You can only back up the system state data on a local computer. You cannot back up the system state data on a remote computer.
- You can also use the Backup wizard to back up system state data.

If you are backing up the system state data to a tape, and the Backup program indicates that there is no unused media available, you might have to use Removable Storage to add your tape to the Backup media pool. For more information about Removable Storage, see "Data Storage and Management" in this book.

## Boot and System Files

Backup considers the functionality of WFP when backing up and restoring boot and system files. System files are backed up and restored as a single entity. The Window File Protection service catalog file, located in the folder %SystemRoot%\system32\ catroot\{F750E6C3-38EE-11D1-85E5-00C04FC295EE}, is backed up with the system files.

In Windows NT 4.0 and earlier, backup programs could selectively backup and restore operating system files as they would data files, allowing for incremental backup and restore operations of most operating system files. Windows 2000, however, does not allow incremental restoration of operating system files.

The Advanced Backup option **Automatically back up system protected files with the System State** backs up all system files that are in your %SystemRoot% directory, in addition to the startup files that are included with the system state data.

## Active Directory

A directory service provides a way to locate and identify users and resources available in a distributed system. Windows 2000 Server includes Active Directory to manage the enterprise directory information for distributed networks. The information in this directory must be considered within your overall backup and restore strategy.

The directory service can be backed up either online or offline, but the restoration process can be completed only when the directory is offline, similar to Microsoft® Exchange Server. The Active Directory database can only be backed up completely; incremental backup is not supported.

Directory information can be replicated, so the system administrator needs to formulate a recovery plan. One option is to restore a replica of the directory and then propagate changes that occurred since the backup from other replicas in the domain. The domain controller can also be restored from tape backup. This assumes that other replicas are not corrupted. If the other replicas are corrupted, you might need to do an authoritative restore. For more information about Active Directory backups, see the *Distributed Systems Guide*.

## Certificate Services Information

Windows 2000 includes a certificate server service that can be installed and configured to provide X.509 certificates for clients. A full backup of the Certificate Services database must exist before it can be restored.

## Cluster Database

System state data includes the cluster database. For more information about backing up and restoring clusters, see the *Distributed Systems Guide*.

## Registry

The contents of the registry are backed up and restored when you back up and restore system state data.

When you back up the system state data, a copy of your registry files is also saved in the folder %SystemRoot%\Repair\Regback. If your registry files become corrupted or are accidentally erased, you can use these files to repair the registry without performing a full restore of the system state data. This method of repairing the registry is only recommended for advanced users.

---

**Important** It is important that you have current, reliable backup copies of the registry. Make sure to back up the registry as part of your regular backup routine and before you edit the registry. If you select the **System State** check box on the **Backup** tab in Backup, the registry is automatically backed up.

---

## Performance Counter Configuration

The performance counter configuration files are also backed up and restored as part of the boot data.

## Component Services Class Registration Database

The Component Object Model (COM) is a binary standard for writing component software in a distributed systems environment. The Component Services Class Registration Database is backed up and restored with the system state data.

Component Services have special backup and restore considerations. There are two Component Services elements on each system: the component binaries, including dynamic-link libraries (DLLs) and executable files (EXEs), and the Component Services database. The components are backed up as a part of normal file enumeration. The Component Services database, however, is backed up and restored as a part of the system state data.

# Using Backup

Backup is a graphical tool used with a variety of storage media to back up and restore files on either NTFS or file allocation table (FAT) volumes. Backup also simplifies archiving and allows you to use the Windows 2000 Job Scheduler for automating backup jobs.

Tasks such as mounting and dismounting a tape or disk are done by Removable Storage. It tracks and controls backup media, usually organized into pools, on storage devices, and allows applications such as Backup to share robotic changers and media libraries. Once started, Removable Storage is transparent to the data path, so you only need to access it when changing media, not when performing a backup or restore operation.

**Note** You can back up to a file or a tape without using Removable Storage as long as the backup medium is not part of a media pool.

## Selecting Backup Type

There are several factors to consider when deciding what backup to use, including the following issues:

- The normal backup type is best when a large amount of data changes between backups or to provide a baseline for the other backup types.
- The incremental backup type is best to record the progression of frequently changed data.
- The differential backup type simplifies the process for restoring files.
- To provide for long-term storage with fewer media, you can use a combination of a normal backup plus either incremental or differential backups.

Some backup types use backup markers, also known as archive attributes, to track when a file has been backed up. When the file changes, Windows 2000 marks the file to be backed up again. Files or directories that have been moved to a new location are not marked for backup. Backup allows you to choose to back up only files with this marker set, and you can choose whether or not to mark files as having been backed up.

Table 12.3 lists the main advantages and disadvantages associated with common types of backup.

**Table 12.3 Advantages and Disadvantages of Backup Types**

| Backup type | Advantages | Disadvantages |
|---|---|---|
| Normal | Files are easy to find because they are always on a current backup of your system or on one medium.<br>File recovery requires only one medium or set of media. | Most time-consuming.<br>If files do not change frequently, backups are redundant. |
| Incremental | Least data storage space required.<br>Least time-consuming. | Files can be difficult to find, because they can be on several media. |
| Differential | Recovery requires only the last normal backup medium and last differential medium.<br>Less time-consuming than normal backups. | Recovery takes longer than if files were on a single medium.<br>If large amounts of data change daily, backups can consume more time than incremental backups. |

## Selecting Backup Media

Backup displays a list of all available storage devices in **Backup** destination the **Backup** tab. If no external devices are detected, you can back up to disk. If you want to back up to media not managed by Removable Storage, make sure that the disk or tape is loaded in a stand-alone storage device.

**Note** If you are backing up to media managed by Removable Storage, you must first start Removable Storage in the Services snap-in. To back up to new media, you must first make the media available in the media scratch pool. If you want to back up to an existing media pool, the media must be loaded in the library. For more information about using Removable Storage, see "Data Storage and Management" in this book.

## Files Skipped During Backup

Backup skips the following file types during the backup and restore processes:

- Files that are held open by other applications.
- Files that are skipped by default by Backup.

### Files Skipped by Default

Backup skips certain files by default, including the following:

- Files that you do not have permission to read. Only users with backup rights can copy files that they do not own.
- Files that are temporary in nature, such as Pagefile.sys, Hiberfil.sys, Win386.swp, 386spart.par, Backup.log, and Restore.log. These files are neither backed up nor restored by Backup. The list of skipped files is embedded into Backup and cannot be changed.
- Registry files on remote computers. Windows 2000 backs up only local registry files.

### Locked Files

Windows 2000 lets you back up local files that are exclusively locked by the operating system, such as event logs and registry files; however, Backup skips those files held open by other processes.

## Encrypted Files

If encrypted files exist on backup tapes, it is important to ensure that user keys, particularly the recovery agent keys, are also stored safely on backup media. Certificate Services provides methods to export keys to floppy disks, or other removable media so that they can be secured in a trusted location.

For more information about Encrypting File System (EFS), see "Encrypting File System" in the *Distributed Systems Guide*.

## Backing Up Files on Your Local Computer

Backup lets you back up any file on your local disk system. This section describes various files on your local computer that it is recommended to back up.

### Logon Scripts

Logon scripts are files that can be assigned to user accounts. Each time a user logs on, the assigned logon script is run. You can use logon scripts to configure user working environments by creating network connections and starting applications. Logon scripts are useful when you want to affect the user work environment without managing all aspects of it. A logon script is always downloaded from the computer running Windows 2000 Server that validates a user's logon request.

On Windows 2000 Server domains, any domain controller can authorize a user's logon attempt. To ensure that logon scripts always work, make sure that logon scripts for all user accounts in a domain exist on every domain controller.

To ensure that logon scripts are always available and consistent, use the Active Directory Replicator service. In replication, copies of the files are sent across the network to other computers. A domain controller sends the logon scripts to the other computers in the domain.

---

**Note** Only use replication for critical information that must be available because replicating files puts a large load on the network. For example, use replication for such critical data as the DHCP and WINS databases. For more information, see the *Microsoft® Windows® 2000 Server Resource Kit TCP/IP Core Networking Guide*.

---

## User Data

Most changes on a server occurs in the users' folders. Users constantly add, modify, or delete files from the computer. It is recommended that you back up changes to users' folders daily.

Some users keep most of the files that they want backed up on file servers. Other users require that data on local computers be backed up. Your backup procedures need to consider both situations.

## Application Programs

Network users primarily use applications, such as Microsoft® Word. You can reinstall the executable files from the original distribution media, but the time and productivity lost make this approach less than ideal. Additionally, if you have customized the applications to suit the needs of your organization, then reproducing those settings can be more difficult than reloading the programs themselves. Since the applications rarely change, backing them up as part of your backup procedure ensures that the latest version is always available without using a lot of offline storage space.

## Archived Data

Backup can back up data that has been archived using Remote Storage. If the archived files are on your local disk, the backup operation works the same as any other. If the archived files are offline or remote, only placeholders on your local disk for the archived data are backed up. If you elect to back up the remote archived data, do so by direct media copy (tape to tape).

## Backing Up Files on Remote Computers

You can use Backup to back up files on any computers to which you can connect remotely. This allows a single media drive to be shared across an entire network and one backup policy to be in effect for the entire network.

**Note** Backup does not recognize MS-DOS or computers that are running Windows 3.1.

# Backing Up Windows 2000 Services and NTFS Features

Windows 2000 includes a number of system services and NTFS features that require special handling of descriptive catalog, metadata, or logging data stores. The binaries of these extensions and services are backed up during normal file backup. Temporary files or other files that can be skipped are registered to be excluded.

## Remote Storage Considerations

Implement Remote Storage should be implemented within a well-planned primary backup scheme. Remote Storage media are standard Microsoft Tape Format (MTF) tapes. MTF-compliant backup applications can catalog the tapes and restore migrated files from Remote Storage media.

**Important** Do not confuse Remote Storage media with primary backup media or view Remote Storage as a substitute for primary backups. Remote Storage migrates data that is used infrequently. Busy data that needs to be restored immediately after a primary storage catastrophe, is less likely to be present on Remote Storage media.

Remote Storage enhances system usability by ensuring a steady supply of free space on file server volumes. It also enhances backup procedures because infrequently used data is no longer present on the volume, and this data doesn't need to be included in a full backup. It is not meant to protect data the way that backup applications do.

Where backup and Remote Storage share a single tape changer that only has one tape drive, backups of this type cannot be created because they would be contending for use of the single tape drive. Remote Storage uses Removable Storage to arbitrate its access to secondary storage. If Backup also uses Removable Storage, then under normal conditions the backup application and Remote Storage can share devices. However, Remote Storage recalls cannot preempt backup jobs, so it is necessary to ensure that backups do not cause recalls when only a single tape drive is available.

When writing to tape media, Remote Storage uses the MTF. When files cannot be recovered from regular backups, Backup uses MTF tapes to restore the Remote Storage data from the Remote Storage tapes.

For more information about Remote Storage, see "Data Storage and Management" in this book.

## Backing Up Remote Storage Databases

Remote Storage requires that the Remote Storage database files be backed up during Backup operations. If the database files are corrupted or lost, stop all Remote Storage services, restore the database files, and then restart the service. If the files in %SystemRoot%\system32\NtmsData are backed up, Remote Storage functions properly after they are restored. Make sure that the system directories are assigned to the same path, including drive letter and directory structure. For example, if the system path is D:\Winnt\system32 prior to the recovery of Windows 2000, after recovery operations the drive letter is still D and the path is still Winnt\system32.

## Protecting Remote Storage Media

Administrators can make up to three copies of each Remote Storage medium. This ensures that tape media, which is less durable than other types of media, is available to the administrator for multiple purposes, including rotating media to off-site storage and having data redundancy in case a tape becomes no longer usable. Media sets can be removed from changers to provide shelf storage of infrequently used data. Media sets can also be added to the backup sets to provide a full set of managed data for recovery.

Using the Remote Storage snap-in, the administrator can identify the copy state of each Remote Storage medium, request that copies be created or updated, and request that media be recreated from copies. Media copies should be part of the normal backup procedure for a Remote Storage volume.

A minimum of two drive changers and three tapes of equal length are needed to create media copies.

Once Remote Storage data has been moved to the media, a copy set of all Remote Storage media can be produced.

▶ **To make media copies**

1. In the Remote Storage snap-in, set the number of needed media copies on the general properties tab. You need enough media within the changer to be able to create copies of each medium that is within the Remote Storage media list.
2. In the media context menu, start the Submit Synchronize Media Copies Now wizard. This will submit a one-time-only scheduled job to provide a copy of the media.

---

**Note** The schedule can be updated under the scheduled task manager to provide an update of the copy set regularly. Only use this option if media sets are not being rotated out of the changer.

---

▶ **To add a media copy to the Backup set for recovery**

1. After performing the media copy procedure, right-click the media set that is going to be used as part of the backup set.

   The copy must be in the Remote Storage pool before it can be deleted from the Remote Storage databases. By deleting the media copy, you set the tape to Idle, Available.
2. In the Remote Storage pool, right-click the appropriate media, and then click **Copy**.
3. In the Remote Storage pool, right-click the wanted media pool, and then click **Paste**.

   This allows the media copy to become part of the Backup set. This medium can subsequently be removed from the changer. A medium that is created by Backup does not carry an on tape catalog and requires significantly more time to restore than media produced from Remote Storage. Use multiple media copies to recover the media from a standard media copy. The deleted media copies can be recreated. Copies that are deleted from the database are not available for Remote Storage recovery but can be recovered using Backup.

# Remote Storage Self-Backup

Although Remote Storage requires that managed files be backed up, it also copies critical structured storage and Jet database information on Remote Storage media.

# Exchange Database

Although the Exchange directory and information store services are independent of each other, they are built on top of Jet databases.

## Backing up Exchange

The Exchange services are backed up online with the data divided into two parts: the database, and the transaction logs and patch files. The transaction logs can be backed up independently from the database during an incremental backup. Backup determines which services are running on the specified server.

---

**Note** To restore, the Exchange service needs the database and a sequential list of transaction logs. If circular logging is enabled on Exchange services, it is not possible to perform a no-loss restore. A no-loss restore uses the log files that reside on the system prior to the restore along with the restored log files to bring the system data to a state just before the crash. Thus if circular logging is enabled, then the existing logs must be deleted. After the restore, the system data is at the state it was just after the backup.

If the log file path is not set, a new log is created on the default directory. Otherwise, the existing log files are used.

---

## Restoring Exchange

Exchange services are restored offline. To perform the restoration, Backup stops the associated services. The Win32 Service Manager functions are used to control the services.

# SQL Database Backup

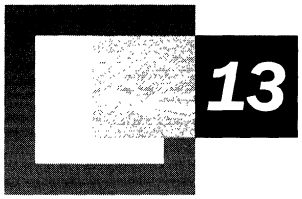
Microsoft® SQL Server™ should not be backed up using Backup. SQL Server has a built-in backup capability. A separate process should be in place to ensure that SQL Server is properly backed up.

# Logical Volume Information

In Windows 2000, the Fault Tolerant Disk state is stored on disk in hidden sectors. This is a change from Windows NT, which stored this information in the registry. Backup applications do not save or restore Fault Tolerant Disk configuration information in Windows 2000. It is up to the operator to reconfigure the disks to the original configuration (including stripe sets, mirrors, and volume sets) prior to restore.

CHAPTER 13

# Repair, Recovery, and Restore




No computer is failure proof, but preventative measures and strategic planning can make a computer running Microsoft® Windows® 2000 more failure resistant. Developing plans and procedures for recovering from failures before they occur can also minimize damage and time lost. Maintaining records about your hardware and software configurations, and regularly backing up data and system configurations can help you recover from serious failures.

**In This Chapter**

Repairing a Windows 2000 Installation 527
Restoring System State Data 544
Restoring the Registry 545
Restoring Data 546
Recovering Your Disk Configuration 546
Recovering a Mirrored or RAID-5 Volume 547
Recovering Data in Remote Storage 551

### Related Information in the Resource Kit

- For more information about planning before failures occur, see "Planning a Reliable Configuration" in this book.
- For more information about creating backups to use during recovery, see "Backup" in this book.
- For more information about common disk problems, see "Disk Concepts and Troubleshooting" in this book.

# Repairing a Windows 2000 Installation

Windows 2000 provides you with the option of fixing problems by using either the Recovery Console or the Emergency Repair Process. During Setup you are asked if you want to set up Windows 2000, repair a Windows 2000 installation, or quit Setup. When you select the Repair option by pressing R, the following information is displayed on your screen:

```
Windows 2000 repair options:
   To repair a Windows 2000 installation by using
   the recovery console, press C.
   To repair a Windows 2000 installation by using
   the Emergency Repair Process, press R.
```

Using each of these options to repair a Windows 2000 installation is discussed in the following sections.

---

**Important** If the repair options do not work, it might be necessary to perform an in-place upgrade. For more information about this option, see "If the Emergency Repair Process Does Not Fix Your System" later in this chapter.

---

## Using the Recovery Console

The Recovery Console is a text-mode command interpreter that is separate from the Windows 2000 command prompt and allows the system administrator to gain access to the hard disk of a computer running Windows 2000, regardless of the file format used, for basic troubleshooting and system maintenance. Since starting Windows 2000 is not a prerequisite for using the Recovery Console, it can help you recover when your Windows 2000–based computer does not start properly or at all. The Recovery Console allows you to obtain limited access to NTFS, file allocation table (FAT), (FAT16 and FAT32) volumes without starting the graphical interface. The Recovery Console allows administrators and Microsoft Product Support Services technicians to start and stop services, and repair the system in a very granular way. It can also be used to repair the master boot record (MBR) and boot sector and to format volumes. The Recovery Console prevents unauthorized access to volumes by requiring the user to enter the system administrator password.

---

**Note** If you are using software mirroring or if you are running the Recovery Console using a Remote Install Server, see the Microsoft Knowledge Base link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

---

## Starting the Recovery Console

To start the Recovery Console, start the computer from the Windows 2000 Setup CD or the Windows 2000 Setup floppy disks. If you do not have Setup floppy disks and your computer cannot start from the Windows 2000 Setup CD, use another Windows 2000–based computer to create the Setup floppy disks. For more information about creating the Windows 2000 Setup floppy disks, see "Planning a Reliable Configuration" in this book or Windows 2000 Server Help.

If the Recovery Console was installed on the local hard disk, it can also be accessed from the Windows 2000 startup menu. However, if the MBR or the system volume boot sector has been damaged, you need to start the computer using either the Windows 2000 Setup floppy disks or the Windows 2000 Setup CD to access the Recovery Console.

To add the Recovery Console to existing installations of Windows 2000, on the **Start** menu, click **Run**, and then type:

```
F:\I386\Winnt32.exe /cmdcons
```

where **F** is the CD-ROM drive letter.

This installation requires approximately 7 megabytes (MB) of disk space on your system partition.

---

**Important** You cannot preinstall the Recovery Console on a computer that contains a mirrored volume. First break the mirror, then install the Recovery Console. After the Recovery Console is installed, you can re-establish the mirrored volume. For more information, see the Microsoft Knowledge Base link on the Web Resources page at http://windows.microsoft.com/windows2000/reskit/webresources.

---

Start the computer and enter Windows 2000 Setup. Press ENTER at the "Setup Notification" screen. Press R to repair a Windows 2000 installation, and then press C to use the Recovery Console.

When the Recovery Console is started, the following information is displayed:

```
Microsoft Windows 2000(TM) Recovery Console.

The Recovery Console provides system repair and recovery
functionality.

Type EXIT to quit the Recovery Console and restart the computer.

1: C:\WINNT

Which Windows 2000 installation would you like to log onto
(To cancel, press ENTER)?
```

---

**Important** If there is more than one installation of Windows 2000 or Microsoft® Windows NT® 4.0 or earlier, they are also shown in the Recovery Console startup menu.

Mirrored volumes appear twice in the Recovery Console startup menu, but each entry has the same drive letter so they are actually the same drive.

Changes made in the Recovery Console to mirrored volumes are mirrored.

---

To access the disk by using the Recovery Console, press the number key representing the Windows 2000 installation that you want to repair, and then press ENTER. The Recovery Console then prompts you for the administrator password. If you press ENTER without typing a number, the Recovery Console exits and restarts the computer.

---

**Note** To use the Recovery Console, you must know the password for the local Administrator account. If you do not have the correct password, Recovery Console does not allow access to the computer. If an incorrect password is entered three times, the Recovery Console quits and restarts the computer. However, you can use either the Group Policy snap-in or the Security Configuration and Analysis snap-in to specify automatic administrative logon.

---

Once the password has been validated, you have full access to the Recovery Console, but limited access to the hard disk. You can only access the following folders on your computer:

- %SystemRoot%. If you have multiple Windows installations, this is on the partition that contains Boot.ini and other Windows files required to start the system.

- %Windir% and subfolders of the Windows 2000 installation that you are currently logged on to.
- %SystemRoot%\Cmdcons and its subfolders.

---

**Note** With the **set** command enabled, you can copy files to removable media and access all paths on the system. The **set** command is an optional Recovery Console command that can be enabled by either the Group Policy snap-in or the Security Configuration and Analysis snap-in. For more information about the **set** command, see "Supported Commands" later in this chapter.

---

The Recovery Console prevents access to other folders such as Program Files or Documents and Settings, as well as to folders containing other installations of Windows 2000. However, you can use the **logon** command to access an alternate installation. Alternatively, you can gain access to other installation folders by restarting the Recovery Console, choosing the number representing that installation, and then entering the administrator password for that installation.

You cannot copy a file from the local hard disk to a floppy disk, but you can copy a file from a floppy disk or a CD-ROM to any hard disk, and from a hard disk to another hard disk. However, with the **set** command enabled, you can copy files to a floppy disk. The Recovery Console displays an "Access is denied" error message when it detects invalid commands.

---

**Important** The **set** command makes use of Recovery Console environment variables to enable, among other options, disk write access to floppy disks. To enable the user to modify the restricted default Recovery Console environment variables, a policy setting must be made. For more information about enabling the **set** command in Recovery Console, see the procedure later in this section.

---

The Recovery Console buffers previously entered commands and makes them available to the user with the up and down arrow keys. To edit a previously entered command, use **backspace** to move the cursor to the point of the edit and retype the remainder of the command. At any point, you can quit the Recovery Console and restart the computer by typing **exit** at the command prompt.

---

**Important** The Recovery Console might not map disk volumes with the same drive letters that they have in Windows 2000. If you are having trouble copying files from one location to another, use the **map** command from the Recovery Console to make sure that the drive mappings for both the source and the target locations are correct.

Several of the Recovery Console commands are not fully functional to users who have converted to dynamic disks. For more information about dynamic disks, see the chapter "Disk Concepts and Troubleshooting" in this book.

---

## Supported Commands

Table 13.1 lists the commands that are supported by the Recovery Console. You can use the **help** command to list the commands supported by the Recovery Console.

**Note** The switch **/?** works with every Recovery Console command to display a help screen offering a description of the command, its syntax, definitions of its parameters, and other useful information.

**Table 13.1 Available Recovery Console Commands**

| Command | Explanation |
|---|---|
| attrib | Changes attributes on one file or directory.<br>`ATTRIB -R \| +R \| -S \| +S \| -H \| +H \| -C \| +C filename`<br><br>`+ Sets an attribute.`<br>`- Clears an attribute.`<br>`R Read-only file attribute.`<br>`S System file attribute.`<br>`H Hidden file attribute.`<br>`C Compressed file attribute.`<br><br>More than one attribute can be set or cleared at a time. To view attributes, use the **dir** command. |
| batch | Executes commands specified in a text file.<br>`BATCH Inputfile [Outputfile]`<br><br>`Inputfile Specifies the text file that contains the list of commands to be executed.`<br>`Outputfile If specified, contains the output of the specified commands. If not specified, the output is displayed on the screen.`<br><br>Batch cannot be one of the commands included in the Inputfile. |

*continued*

**Table 13.1 Available Recovery Console Commands *(continued)***

| Command | Explanation |
|---|---|
| cd/chdir | Displays the name of the current directory, or switches to a new directory.<br>`CHDIR [path]`<br>`CHDIR [..]`<br>`CHDIR [drive:]`<br>`CD [path]`<br>`CD [..]`<br>`CD [drive:]`<br><br>`CD ..` Specifies that you want to change to the parent directory.<br><br>Type **CD [drive:]** to display the current directory in the specified drive. Type **CD** without parameters to display the current drive and directory.<br>The **chdir** command treats spaces as delimiters. Use quotation marks around a directory name containing spaces. For Example:<br>**cd "\winnt\profiles\username\programs\start menu"**<br>**Chdir** operates only within the system directories of the current Windows installation, removable media, the root directory of any hard disk partition, or the local installation sources. |
| chkdsk | Checks a disk and displays a status report,<br>`chkdsk [drive:] [/p] \| [/r]`<br><br>`[drive:]` Specifies the drive to check.<br>`/p` Check even if the drive is not flagged dirty. bad.<br>`/r` Locates bad sectors and recovers readable information (implies /p).<br><br>**Chkdsk** may be used without any parameters, in which case the current drive is checked with no switches. You can specify the listed switches.<br>**Chkdsk** requires the Autochk.exe file. **Chkdsk** automatically locates Autochk.exe in the startup (boot) directory. If it cannot be found in the startup directory, **chkdsk** attempts to locate the Windows 2000 Setup CD. If the installation CD cannot be found, **chkdsk** prompts for the location of Autochk.exe. |
| cls | Clears the screen. |

*continued*

**Table 13.1 Available Recovery Console Commands *(continued)***

| Command | Explanation |
|---|---|
| copy | Copies a single file to another location.<br>`copy source [destination]`<br><br>`source` Specifies the file to be copied.<br>`Destination` Specifies the directory and/or file name for the new file.<br><br>The source might be removable media, any directory within the system directories of the current Windows installation, the root of any drive, the local installation sources, or the cmdcons directory.<br><br>The destination might be any directory within the system directories of the current Windows installation, the root of any drive, the local installation sources, or the cdirectory. The destination cannot be removable media. If a destination is not specified, it defaults to the current directory. **Copy** does not support replaceable parameters (wild cards). **Copy** prompts if the destination file already exists. A compressed file from the Windows 2000 Setup CD is automatically decompressed as it is copied. |
| del/delete | Deletes one file.<br>`del [drive:][path]filename`<br>`delete [drive:][path]filename`<br><br>`[drive:][path]filename` Specifies the file to delete.<br><br>**Delete** only operates within the system directories of the current Windows installation, removable media, the root directory of any hard disk partition, or the local installation sources.<br><br>**Del** and **delete** do not support replaceable parameters (wild cards). |
| dir | Displays a list of files and subdirectories in a directory.<br>`dir [drive:][path][filename]`<br><br>`[drive:][path][filename]` Specifies drive, directory, and/or files to list.<br><br>**Dir** lists all files, including hidden and system files.<br><br>Files might have the following attributes:<br>`a` Files ready for archiving<br>`c` Compressed<br>`d` Directory<br>`e` Encrypted<br>`h` Hidden<br>`p` Reparse Point<br>`r` Read-only<br>`s` System file |

*continued*

**Table 13.1 Available Recovery Console Commands *(continued)***

| Command | Explanation |
|---|---|
| disable | Disables a Windows system service or driver.<br>`disable servicename`<br><br>`servicename` The name of the service or driver to be disabled.<br><br>**Disable** prints the old start_type of the service before resetting it to SERVICE_DISABLED. You should make a note of the old start_type, in case you need to enable the service again.<br>The start_type values that the **disable** command displays are:<br>`SERVICE_DISABLED`<br>`SERVICE_BOOT_START`<br>`SERVICE_SYSTEM_START`<br>`SERVICE_AUTO_START`<br>`SERVICE_DEMAND_START` |
| diskpart | Manages the partitions on your hard disk volumes.<br>`diskpart[/add \| /delete] [device-name \| drive-name \| partition-name] [size]`<br><br>`/add` Create a new partition<br>`/delete` Delete an existing partition<br><br>`device-name` Device name for creating a new partition (such as \Device\HardDisk0)<br>`drive-name` Drive-letter based name for deleting an existing partition (such as D:)<br>`partition-name` Partition-based name for deleting an existing partition and can be used in place of the drive-name argument (such as \Device\HardDisk0\Partition1)<br>`size` Size of the new partition, in megabytes<br><br>If no arguments are used, a user interface for managing your partitions appears.<br>**Warning**<br>This command can damage your partition table if the disk has been upgraded to dynamic disk. Always use Disk Management to modify the structure of dynamic disks. |

*continued*

**Table 13.1 Available Recovery Console Commands *(continued)***

| Command | Explanation |
|---|---|
| enable | Enables a Windows system service or driver.<br>`enable servicename [start_type]`<br><br>`servicename` Name of the service or driver to be enabled.<br>`start_type` How the service or driver is scheduled to be started. Valid start-type values are:<br><br>SERVICE_BOOT_START<br>SERVICE_SYSTEM_START<br>SERVICE_AUTO_START<br>SERVICE_DEMAND_START<br><br>**Enable** prints the old start_type of the service before resetting it to the new value. Note the old value, in case it is necessary to restore the start_type of the service. If you do not specify a new start_type, **enable** prints the old start_type. |
| exit | Quits the Recovery Console and restarts your computer. |
| expand | Expands a compressed file.<br>`EXPAND source [/F:filespec] [destination] [/Y]`<br>`EXPAND source [/F:filespec] /D`<br><br>`source` Specifies the file to be expanded. May not include wildcard (* and ?) characters.<br>`Destination` Specifies the directory for the new file. Default is the current directory.<br>`/y` Do not prompt before overwriting an existing file.<br>`/f:filespec` If the source contains more than one file, this parameter is required to identify the specific file(s) to be expanded. May include wildcards.<br>`/d` Do not expand; only display a directory of the files which are contained in the source.<br><br>The destination might be any directory within the system directories of the current Windows installation, the root of any drive, the local installation sources, or the Cmdcons directory. The destination cannot be removable media. The destination file cannot be read-only. Use the **attrib** command to remove the read-only attribute. **Expand** prompts if the destination file already exists unless **/Y** is used. |

*continued*