

Performance Best Practices and Benchmarking Guidelines

Workstation 6.5



Performance Best Practices and Benchmarking Guidelines

Item: EN-000007-00

You can find the most up-to-date technical documentation on the VMware Web site at:

<http://www.vmware.com/support/>

The VMware Web site also provides the latest product updates.

If you have comments about this documentation, submit your feedback to:

docfeedback@vmware.com

© 2007-2008 VMware, Inc. All rights reserved. Protected by one or more U.S. Patent Nos. 6,397,242, 6,496,847, 6,704,925, 6,711,672, 6,725,289, 6,735,601, 6,785,886, 6,789,156, 6,795,966, 6,880,022, 6,944,699, 6,961,806, 6,961,941, 7,069,413, 7,082,598, 7,089,377, 7,111,086, 7,111,145, 7,117,481, 7,149,843, 7,155,558, 7,222,221, 7,260,815, 7,260,820, 7,269,683, 7,275,136, 7,277,998, 7,277,999, 7,278,030, 7,281,102, 7,290,253, 7,356,679, 7,409,487, 7,412,492, 7,412,702, and 7,424,710; patents pending.

VMware, the VMware “boxes” logo and design, Virtual SMP, and VMotion are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

About This Book 5

- Terminology 5
- Intended Audience 6
- Document Feedback 6
- Technical Support and Education Resources 6
 - Online and Telephone Support 6
 - Support Offerings 6
 - VMware Professional Services 6

1 VMware Workstation Performance 7

- Hardware for VMware Workstation 8
 - CPUs for VMware Workstation 8
 - Hardware Paging Support for Virtualization 8
 - Memory for VMware Workstation 8
 - Disks for VMware Workstation (General) 8
 - Disks for VMware Workstation (SAN, NAS, and RAID) 8
 - Networking for VMware Workstation 9
- Host Operating System Configuration 10
 - Memory and the Host Operating System 10
 - Disks and the Host Operating System 10
 - Networking and the Host Operating System 10
 - Services and Settings in the Host Operating System 11
 - Disable Unneeded Services 11
 - Avoid Non-Default Configuration Settings 11
- VMware Workstation and Virtual Machines 12
 - CPU Considerations in Workstation and Virtual Machines 12
 - CPU Resources 12
 - Configuring Process Scheduling on Windows Hosts 12
 - Configuring Workstation for Hardware Paging Support for Virtualization 13
 - Memory Resources in Workstation and Virtual Machines 13
 - Reserving Memory for Workstation 13
 - Specifying the Amount of Memory That Can Be Swapped to Disk 13
 - Memory Allocated to Virtual Machines 14
 - Memory Trimming 14
 - Page Sharing 14
 - Hardware Paging Support for Virtualization 14
 - Disk Considerations in Workstation and Virtual Machines 15
 - Networking Considerations in Workstation and Virtual Machines 15
 - Debug and Logging Modes in Workstation 15
 - Turn off Debug Mode in Workstation 15
 - Logging in Workstation 16
 - Other Workstation and Virtual Machine Considerations 16
- Guest Operating Systems 18
 - Guest Operating System General Considerations 18
 - Running Paravirtualized Operating Systems 18
 - CPU Considerations in Guest Operating Systems 18
 - Guest Operating System UP Versus SMP HAL/Kernel 18

Guest Operating System Idle Loops	19
Timer Interrupt Rates	19
Memory Considerations in Guest Operating Systems	20
Disk Considerations in Guest Operating Systems	21
Defragment Disks	21
Performance Impact of Defragmenting Snapshots and Linked Clones	21
Networking Considerations in Guest Operating Systems	22
Software Version Considerations in Guest Operating Systems	22
Guest Operating System Settings	23
2 Benchmarking VMware Workstation	25
“Apples-to-Apples” Benchmarking Comparisons	25
Benchmarking Design	26
General Benchmarking Methodology	26
Timing Considerations When Benchmarking	27
Benchmarking Tools	28
CPU-Related Benchmarks	28
Memory-Related Benchmarks	28
Disk-Related Benchmarks	28
Networking-Related Benchmarks	28
Application Benchmarks	28
Multiple-Virtual Machine Consolidation Benchmarks	28
Discouraged Benchmarks	28
Performance Monitoring Tools	29
Windows: Perfmon	29
Linux: vmstat	29
Linux: sar	29
Hardware for Benchmarking	30
General Hardware Considerations When Benchmarking	30
CPU Considerations When Benchmarking	30
Memory Considerations When Benchmarking	30
Disks (SAN, NAS, and RAID) and Benchmarking	30
Networking Considerations When Benchmarking	30
Other Device Considerations When Benchmarking	31
Configuring and Running VMware Workstation for Benchmarking	32
Memory Trimming and Page Sharing in Workstation	32
Use Purpose-Built Virtual Machines When Benchmarking	32
Guest Operating Systems	33
CPU Factors in Guest Operating Systems	33
Guest Operating System UP Versus SMP HAL/Kernel	33
32-bit Versus 64-bit CPUs	33
Disk Considerations and Guest Operating Systems	33
Other Guest Operating System Considerations	33
Glossary	35

About This Book

This book provides guidance in obtaining the best possible performance using VMware® Workstation 6.5, both in a production environment and when running benchmarks. Though this book focuses on Workstation 6.5, most of the guidelines presented here also apply to prior versions of VMware Workstation, as well as to VMware Server and VMware Player.

This document consists of the following parts:

- [Chapter 1, “VMware Workstation Performance,”](#) on page 7, addresses performance tuning and best practices for the host system, the VMware software, and the guest systems.
- [Chapter 2, “Benchmarking VMware Workstation,”](#) on page 25, describes performance benchmarking methodology as well as some areas in which tuning for a benchmark test can be different than tuning for performance.
- A [“Glossary,”](#) on [page 35](#), completes this guide.

There are some areas in which the best-performing configurations of VMware Workstation virtual machines vary slightly from the configurations of native machines. One of the goals of this book is to provide guidance about these variations. To this end, we discuss configuration of the host, the VMware Workstation software, and the operating systems and applications in the individual virtual machines.

The benchmarking guidelines in this book are intended to assist in the acquisition of meaningful, accurate, and repeatable benchmarking results. These guidelines are useful when you perform comparisons between VMware Workstation and either native systems or other virtualization products, as well as when you run benchmarks in virtual machines in general. The guidelines should not, however, be considered VMware best practices in all cases. Among other reasons, this is because benchmarking sometimes involves saturating one resource while overprovisioning others, something that would not be desirable in a production environment.

The book also includes examples of common pitfalls that can impact performance or affect the accuracy or relevance of the results obtained by benchmarking tests.

Terminology

Throughout this book we use the terms *native*, *host*, and *guest*. Brief definitions are included here. More information about these terms, as well as about the other terms in italic font throughout this book, may be found in the [“Glossary”](#) on page 35.

A *native system* is a computer running a single operating system, and on which the applications run directly in that operating system.

A *host system* is a computer on which VMware Workstation software is running.

A *host operating system* is an operating system running directly on a host computer. VMware Workstation runs within the host operating system.

A *guest operating system* is an operating system that runs inside a virtual machine.

Intended Audience

This book is intended for anyone who wants to maximize the performance of VMware Workstation. The information in this book is written for experienced Windows or Linux system administrators who are familiar with virtual machine technology and datacenter operations.

Document Feedback

If you have comments about this documentation, submit your feedback to docfeedback@vmware.com

Technical Support and Education Resources

The following sections describe the technical support resources available to you. To access the current versions of this book and other books, go to:

<http://www.vmware.com/support/pubs>.

Online and Telephone Support

Use online support to submit technical support requests, view your product and contract information, and register your products. Go to:

<http://www.vmware.com/support>

Customers with appropriate support contracts should use telephone support for the fastest response on priority 1 issues. Go to:

http://www.vmware.com/support/phone_support.html

Support Offerings

Find out how VMware support offerings can help meet your business needs. Go to:

<http://www.vmware.com/support/services>

VMware Professional Services

VMware Education Services courses offer extensive hands-on labs, case study examples, and course materials designed to be used as on-the-job reference tools. Courses are available onsite, in the classroom, and live online. For onsite pilot programs and implementation best practices, VMware Consulting Services provides offerings to help you assess, plan, build, and manage your virtual environment. To access information about education classes, certification programs, and consulting services, go to:

<http://www.vmware.com/services/>.

VMware Workstation Performance

This chapter provides guidance in tuning a VMware Workstation 6.5 system for the highest possible performance.

It is separated into the following sections:

- [“Hardware for VMware Workstation”](#) on page 7.
- [“Host Operating System Configuration”](#) on page 9.
- [“VMware Workstation and Virtual Machines”](#) on page 11.
- [“Guest Operating Systems”](#) on page 16.

Hardware for VMware Workstation

This section provides guidance about selecting and configuring hardware for use with VMware Workstation.

CPUs for VMware Workstation

This section addresses CPU considerations when selecting hardware for use with VMware Workstation.

Hardware Paging Support for Virtualization

Hardware paging support for virtualization—also called rapid virtualization indexing (RVI), extended page tables (EPT), or hardware virtual memory management unit (hardware virtual MMU)—is a hardware feature included in some recent processors that addresses the overheads due to memory management unit virtualization. This feature is supported beginning with Workstation 6.5.

Guest operating systems use page tables to maintain guest virtual memory to guest physical memory address mappings. Without hardware paging support for virtualization, Workstation maintains “shadow page tables” that directly map guest virtual memory to host physical memory addresses. These shadow page tables are maintained for use by the processor and are kept consistent with the guest page tables. This allows ordinary memory references to execute without additional overhead, since the hardware translation lookaside buffer (TLB) will cache direct guest virtual memory to host physical memory address translations read from the shadow page table. However, extra work is required to maintain the shadow page table structures.

Hardware paging support for virtualization allows an additional level of page tables that map guest physical memory to host physical memory addresses, eliminating the need for Workstation to intervene to virtualize the MMU in software.

For information about configuring the way Workstation uses hardware paging support for virtualization, see [“Configuring Workstation for Hardware Paging Support for Virtualization”](#) on page 12.

Memory for VMware Workstation

Make sure the host has more physical memory than the total amount of memory that will be used by all of the following:

- The host operating system
- Any applications that will be running natively on the host
- The sum of the memory that will be used by the guest operating systems and the applications on all the virtual machines that will be running at any one time

Disks for VMware Workstation (General)

- For best performance, make sure the host hard drive is large enough that it will have an ample amount of free space. Performance can degrade considerably when VMware Workstation has to use a nearly-full host hard drive when working with snapshots or writing to redo logs.

Disks for VMware Workstation (SAN, NAS, and RAID)

Storage performance issues are often the result of storage hardware mis-configuration. Refer to the manuals provided by your storage vendor for tuning methods and parameters. Here are some of the areas to consider:

- When possible, make sure the read and write caches on SAN and NAS storage devices are enabled and configured to the appropriate sizes. Caches that are either disabled or too small could have an adverse effect on performance.
- Make sure that the queue depth on the host bus adapter is configured appropriately, as this configuration can significantly affect performance (refer to the documentation supplied by your storage vendor for details).
- RAID storage systems come in various configurations (for example, RAID level 0, 1, 2, 3, 4, 5, 6, 0+1, and so on), with different trade-offs for performance and reliability. If using RAID, make sure you have made the appropriate choice for your usage. For example, RAID 0 can provide better bandwidth than RAID 5, but at the cost of decreased fault tolerance.
- If using Fibre Channel storage, ensure that your connection bandwidth is as expected (that is, 1Gbps, 2Gbps, or 4Gbps). To determine the storage connection bandwidth, refer to the management tool provided by your SAN vendor (EMC Navisphere, for example).

Networking for VMware Workstation

- Use network switches instead of hubs.
- Make sure that all the networking infrastructure is appropriately rated. For example, when connecting systems containing Gigabit network interface cards, make sure to use Gigabit switches and Gigabit-rated cables.
- Don't have more physical network interface cards than absolutely necessary. This avoids the unnecessary overhead associated with processing broadcast packets, protocol control packets, and so forth.

Host Operating System Configuration

This section provides guidance regarding configuring the host operating system used with VMware Workstation, as well as other settings that are made from within the host operating system.

Memory and the Host Operating System

- Make sure memory is sized so as to avoid excessive page faults in the host operating system. Information about measuring page faults can be found in “[Memory Considerations in Guest Operating Systems](#)” on page 18.

Disks and the Host Operating System

- On Windows systems the Disk Properties tab associated with each hard disk provides a check box for enabling write caching on the disk and, in some cases, for enabling advanced performance on the disk. Selecting one or both of these check boxes can improve the host’s disk performance in general. Selecting them for the host disks containing VMware virtual disk files can improve VMware disk performance in particular, especially when VMware is making heavy use of the disk.



WARNING Power outage or equipment failure can result in data loss or corruption when write caching or advanced performance are enabled.

- Most versions of Windows can use direct memory access (DMA) for higher performance on IDE hard disks and optical drives (CD or DVD drives). However, this feature might not be enabled by default.

For information about enabling DMA in Windows operating systems, see:

<http://support.microsoft.com/kb/258757>

<http://www.microsoft.com/whdc/device/storage/IDE-DMA.msp>

Networking and the Host Operating System

- If your system has multiple physical network interface cards (in card slots or built-in) make sure you are using the intended network interface cards. If you have more than one network interface card, it is easy to enable and use the wrong one. To avoid this confusion, disable the network interface cards you do not plan to use.
- Most modern network interface cards can operate in multiple modes (such as 10, 100, or 1000Mbps; half duplex or full duplex). Make sure the network interface cards are in full-duplex mode and are configured at their maximum possible bandwidth (that is, 1000Mbps for a Gigabit network interface card).
- Don't change any of the default network interface card driver settings unless there is a valid reason to do so (use the OEM listed recommendations).
- If your network performance is low and your virtual machines are not using *host-only* or NAT networking, you can try disabling the VMware host-only or NAT networking adapters from within the host operating system. For more information about this, see VMware knowledge base article 684, *Slow Network Performance when Using Windows Browsing on the Host*, at:

http://www.vmware.com/support/kb/enduser/std_adp.php?p_faqid=684
- Increasing host NIC interrupt coalescing can improve performance for workloads involving heavy network traffic into the virtual machine. Interrupt coalescing, which is a feature implemented in hardware under driver control on high-performance NICs, allows the operating system kernel to be notified of the reception of a group of network frames through a single hardware interrupt.

Services and Settings in the Host Operating System

This section describes services and settings in the host operating system.

Disable Unneeded Services

Disable any services on the host system that you do not need.

In Windows

- To see all currently running processes, bring up **Windows Task Manager** and click the **Processes** tab.
- To enable or disable services, choose **Control Panel > Administrative Tools > Services**.
- To see programs that are launched on startup from the **Start** menu, choose **Start > Programs > Startup**.
- To add or remove programs that are launched on startup from the **Start** menu, do the following:

In Windows 2000: Right-click within the tool bar, select **Properties**, and click the **Advanced** tab in the **Taskbar and Start Menu Properties** window. Click the **Advanced** button, select **Start Menu > Programs > Startup**. Perform this procedure for the **Start** menu under both your user name and **All Users**.

In Windows XP or Windows 2003: Right-click within the tool bar, select **Properties**, and click the **Start Menu** tab in the **Taskbar and Start Menu Properties** window. Click the **Customize** button, then the **Advanced** button. Select **Start Menu > Programs > Startup**. Perform this procedure for the **Start** menu under both your user name and **All Users**.

In Linux

- To see all currently running processes, run:
`ps auxww` or `top`
- To see all system services, run:
`chkconfig --list`

The `chkconfig` command provides a command-line interface for administering the system services that are invoked from `/etc/rc[0-6].d`

- To add or remove system services, run:
`chkconfig service-name [on|off|reset]`

For example, `chkconfig atd off` and `chkconfig crond off` turn off the `at` and `cron` daemons so that no scheduled jobs can run during the experiment.

Avoid Non-Default Configuration Settings

- Check that any non-default system configuration settings are intentional.

VMware Workstation and Virtual Machines

This section provides guidance regarding the VMware Workstation software itself and the virtual machines that run in it.

CPU Considerations in Workstation and Virtual Machines

This section addresses CPU considerations in Workstation and in virtual machines.

CPU Resources

Try not to overcommit CPU resources:

- Avoid running a dual-processor virtual machine on a single-processor host system, even if the single-processor host has *hyper-threading* (that is, two logical CPUs).
- Avoid running two or more single-processor virtual machines on a single-processor host system, even if the single-processor host has hyper-threading.
- Make sure that CPU resources are not overcommitted on the host system, remembering that virtualization has its own CPU overhead.

Configuring Process Scheduling on Windows Hosts

By default, the processes within a virtual machine contend equally for resources with all other processes running on the host, regardless of whether or not the mouse and keyboard are grabbed by the virtual machine.

VMware Workstation on a Windows host provides the following configuration options to adjust scheduling priorities:

- On a global basis, for all virtual machines. Use the Workstation preference editor, as follows:
 - a Choose **Edit > Preferences**.
 - b Select the **Priority** tab.
- On a per-virtual-machine basis, for virtual machines on which you want to override the default global setting. Use the virtual machine settings editor, as follows:
 - a Select the desired virtual machine.
 - b Choose **VM > Settings**.
 - c On the **Options** tab, select **Advanced**.

These menu items allow you to specify either high or normal priority when the mouse and keyboard are grabbed by the virtual machine and either normal or low priority when they are not grabbed. By default, the priority for virtual machine processing is set to **Normal** for both input grabbed and input ungrabbed.

If you have many background processes or applications and you do not care whether they run with fairly low relative priority while Workstation is in the foreground, set **Input grabbed** to **High** and set **Input ungrabbed** to **Normal**. You should see a noticeable performance boost when using one virtual machine while another virtual machine is running or while some other processor-intensive task (a compile, for example) is running in the background.

If your host machine runs too sluggishly when a virtual machine is running in the background, you can direct Workstation to reduce the priority of that virtual machine when that virtual machine does not have control of the mouse and keyboard. To do this, set **Input grabbed** to **Normal** and set **Input ungrabbed** to **Low**.

Configuring Workstation for Hardware Paging Support for Virtualization

For a description of hardware paging support for virtualization, see [“Hardware Paging Support for Virtualization”](#) on page 7.

Hardware paging support for virtualization is supported beginning with Workstation 6.5. Depending on the type of CPU, the guest operating system and version, and the Execution Mode setting, Workstation 6.5 will enable or disable the use of hardware paging support for virtualization for each individual virtual machine. These defaults are based on extensive testing and will work well the majority of the time. If desired, however, the default can be changed for a specific virtual machine. To control the use of hardware paging support for virtualization for a virtual machine:

- 1 With the virtual machine powered down, choose **VM > Settings**.
- 2 On the **Hardware** tab, select **Processors**.
- 3 Under **Execution Mode**, choose the desired setting in the **Preferred mode** window.

Memory Resources in Workstation and Virtual Machines

This section addresses CPU resource considerations in Workstation and in virtual machines.

Reserving Memory for Workstation

Host operating systems do not behave well when they run low on free memory for their own use. When a Windows or Linux host operating system does not have enough memory for its own use, it can exhibit *thrashing* (that is, constant swapping of data between memory and a paging file on disk).

To set the maximum amount of RAM that Workstation is allowed to reserve for all running virtual machines combined, choose **Edit > Preferences** and select the **Memory** tab. The memory used by Workstation includes the memory made available to the guest operating systems plus a small amount of overhead memory associated with running a virtual machine.

To help guard against virtual machines causing the host to thrash, Workstation enforces a limit on the total amount of memory virtual machines can consume. Some memory must be kept available on the host to ensure the host is able to operate properly while virtual machines are running. The amount of memory reserved for the host depends on the host operating system and the size of the host computer’s memory.

Specifying the Amount of Memory That Can Be Swapped to Disk

By default, Workstation limits the number of virtual machines that can simultaneously run based on the amount of memory specified in the application settings. This prevents virtual machines from causing each other to perform poorly.

To allow more or larger virtual machines to run, adjust the amount of virtual machine memory that the host operating system can swap to disk. To change this setting, choose **Edit > Preferences**, select the **Memory** tab, and select one of the following radio buttons:

- **Fit all virtual machine memory into reserved host RAM** – Strictly apply the reserved memory limit set in the top of the panel. This setting imposes the tightest restrictions on the quantity and memory size of virtual machines that can run at a given time. Because the virtual machines are running entirely in RAM, they have the best possible performance.
- **Allow some virtual machine memory to be swapped** – Allow the host operating system to swap a moderate amount of virtual machine memory to disk if necessary. This setting allows you to increase the number or memory size of virtual machines that can run on the host computer at a given time. Unfortunately, it might also result in reduced performance if virtual machine memory must be shifted between RAM and disk.
- **Allow most virtual machine memory to be swapped** – This option can result in lower performance than the other options. It allows the host operating system to swap as much virtual machine memory to disk as it wants. This setting allows you to run even more virtual machines with even more memory than the intermediate setting does.

Memory Allocated to Virtual Machines

Carefully select the amount of virtual memory you allocate to your virtual machines:

- On the one hand, you should allocate enough memory to hold the working set of applications you will run in the virtual machine.
- On the other hand, unlike native systems, on which more physical memory is almost always better, allocating too much virtual memory to a virtual machine can reduce overall system performance, and can even reduce the performance of the overallocated virtual machine. This is because allocating to a virtual machine much more memory than its working set requires can increase memory pressure at the host level (as the host balances VMware Workstation's memory requests against the system's other memory needs).

This performance penalty due to overallocation might be exacerbated if you select the **Fit all virtual machine memory into reserved host RAM** option (as described in [“Specifying the Amount of Memory That Can Be Swapped to Disk”](#) on page 12).

Memory Trimming

VMware Workstation uses memory-trimming to deallocate unused virtual machine memory for the host to reallocate. Trimming usually has little impact on performance, and it might be needed in low-memory situations. However, memory trimming can reduce disk performance in a virtual machine. To disable memory trimming, use one of the following two methods:

- Add “MemTrimRate=0” to the .vmx file for the virtual machine.
- With the virtual machine opened, choose **VM > Settings**.
 - a On the **Options** tab select **Advanced**.
 - b In the Settings section, select **Disable memory page trimming**.

Page Sharing

VMware Workstation uses page-sharing to allow multiple guest memory pages with identical contents to be stored as a single copy-on-write page. Many workloads present opportunities for sharing memory across virtual machines. For example, several virtual machines might be running instances of the same guest operating system, have the same applications or components loaded, or contain common data.

With page sharing, a workload often consumes less memory than it would when running on a physical machine. As a result, the system can efficiently support higher levels of memory overcommitment.

The amount of memory saved by page sharing is highly dependent on workload characteristics. Enabling page sharing for a workload that consists of many nearly identical virtual machines might free up more than 30 percent of memory, whereas doing so for a more diverse workload might result in savings of less than 5 percent of memory.

Page sharing runs as a background activity that scans for sharing opportunities over time. For a fairly constant workload, the amount of memory saved generally increases slowly until all sharing opportunities are exploited.

Although page sharing can decrease host memory usage, it does consume other system resources, potentially including I/O bandwidth. You might want to avoid this overhead with virtual machines for which host memory is plentiful and I/O latency is important. To disable page sharing, add the following line to the .vmx file for the virtual machine:

```
sched.mem.pshare.enable = "FALSE"
```

Hardware Paging Support for Virtualization

Hardware paging support for virtualization is a technique that virtualizes the CPU's memory management unit (MMU). For a description of this feature, see [“Hardware Paging Support for Virtualization”](#) on page 7; for information about configuring the way Workstation uses this feature, see [“Configuring Workstation for Hardware Paging Support for Virtualization”](#) on page 12.

Disk Considerations in Workstation and Virtual Machines

- For the best performance, store virtual disks on local disk drives instead of on network drives. When it is necessary to store virtual disks on network drives, use the fastest available connection.
- VMware Workstation can emulate either IDE or SCSI disks. Virtual SCSI disks perform better than virtual IDE disks, and are therefore recommended.

NOTE While SCSI disks are the default when creating virtual machines for most operating systems, some operating systems, such as 32-bit Windows XP, require that you download a driver from VMware. Other operating systems, such as Windows 2000, will run without the VMware driver, but have better performance when the driver is installed.

See the *VMware Workstation User's Manual* for more information on this topic.

- Use preallocated disks instead of growable disks. When creating a new virtual machine, the disk wizard offers the option **Allocate all disk space now?** Selecting this option preallocates all the space at the time you create the virtual disk. It requires as much physical disk space as the size you specify for the virtual disk. The alternative is a growable disk which starts small and grows as needed. Preallocated disks yield better performance and more reproducible results than growable disks.
- *Independent virtual disks* may be either *persistent* or *nonpersistent*. With persistent disks, changes are immediately and permanently written to the disk. All changes to a persistent disk remain, even if a snapshot is reverted. With nonpersistent disks, changes to the disk are discarded when you power-off or revert to a snapshot. Independent, persistent disks have the highest performance and are recommended for most workloads. You can configure a virtual machine for independent persistent disks as follows:
 - a With the virtual machine opened but powered off, choose **VM > Settings**.
 - b On the **Hardware** tab select **Hard Disk**.
 - c Click the **Advanced** button in the lower right.
 - d In the **Mode** section, be sure **Independent** and **Persistent** are both selected.
- The use of the snapshot feature reduces performance. If you do not use the snapshot feature, run your virtual machine with no snapshots. To be sure a virtual machine has no snapshots:
 - a Choose **VM > Snapshot > Snapshot Manager**.
 - b If you see a snapshot you do not want, select it and click the **Delete** button.

Networking Considerations in Workstation and Virtual Machines

- You should remove or disable any virtual networking devices that are not required for your usage.

Debug and Logging Modes in Workstation

This section addresses the debugging and logging modes in Workstation.

Turn off Debug Mode in Workstation

VMware Workstation has three modes for collecting debugging information: **None** (in which no debugging information is gathered), **Statistics**, and **Full**. When set to **None** the virtual machine runs faster than in the other two modes.

To disable debugging:

- 1 Start Workstation and select the virtual machine.
The virtual machine can be either powered on or powered off.
- 2 Choose **VM > Settings**.
The virtual machine settings editor opens.
- 3 On the **Options** tab select **Advanced**.
- 4 In the **Settings** section, set the **Gather debugging information** control to **None**.
- 5 Click **OK**.

More information about the various debug modes and when to use them can be found in the *VMware Workstation User's Manual*.

Logging in Workstation

- VMware Workstation has logging enabled by default. Although this generates disk I/O, the impact is extremely small, and we recommend that logging always be left enabled. If you want, however, you can disable logging by adding the following line to the `.vmx` file for the virtual machine in question:

```
logging = "FALSE"
```

More information about virtual machine logging, including how to change the location of the log file, can be found in VMware knowledge base article 1028, *Disabling the Workstation Log File or Storing the Log File in a Different Location*, at:

http://www.vmware.com/support/kb/enduser/std_adp.php?p_faaid=1028

- If you do leave logging enabled, make sure that the log files are stored on a local disk (rather than on a network drive).

Other Workstation and Virtual Machine Considerations

- Make sure you are running a general availability (GA) release of VMware Workstation (rather than a beta or debug version).
- Make sure you have selected the correct guest operating system in the virtual machine settings editor. Selecting the wrong guest operating system should not cause a virtual machine to run incorrectly, but it can significantly reduce the virtual machine's performance.

To verify the current selection:

- a Choose **VM > Settings**.
 - b On the **Options** select **General**.
 - c Look in the **Version** window.
Note that to change the selection you must first power off the virtual machine.
- Check that any non-default VMware Workstation configuration settings are intentional.

- Some operating systems, including Windows NT and Windows 98, poll the optical drives (CD or DVD drives) approximately every second to see if a disc is present. This polling allows the operating systems to execute autorun programs. This polling can also cause VMware Workstation to connect to the host optical drive, which can increase CPU usage and can make the virtual machine appear to pause while the drive spins up. To avoid such pauses, remove the optical drive from your virtual machine unless you need it for your application.

Another option is to configure your virtual machine to start with the optical drive disconnected. This way, the virtual machine seems to have a virtual optical drive containing no disc (and VMware Workstation does not connect to your host optical drive).

To make this change:

- a Choose **VM > Settings**.
- b On the **Hardware** tab select **CD-ROM**.
- c Clear the **Connect at power on** check box.

When the virtual machine is configured not to automatically connect to the CD-ROM at power on you can still use the CD-ROM from within the virtual machine by selecting the **Connected** box after the machine is powered on.

- Due to changes in VMware Workstation beginning with version 5.5, full screen and normal (windowed) mode have nearly the same performance. There is therefore no longer any performance reason to choose one mode over the other.

Guest Operating Systems

This section provides guidance regarding the guest operating systems running in virtual machines.

Guest Operating System General Considerations

This section addresses general guest operating system considerations.

Running Paravirtualized Operating Systems

Workstation 6.5 includes support for virtual machine interface (VMI), used for communication between the guest operating system and the hypervisor, thus improving performance and efficiency. Enabling this support will improve the performance of virtual machines running operating systems with VMI by reducing their CPU utilization and memory space overhead (the later being especially true for SMP virtual machines). Even when only some of the virtual machines on a system use VMI, the performance of all virtual machines on that system might benefit due to the hardware resources freed up to allocate elsewhere. There is no performance benefit to enabling VMI for a virtual machine running a non-VMI operating system.

To enable VMI support for a virtual machine in Workstation:

- a Choose **VM > Settings**.
- b On the **Hardware** tab select **Processors**.
- c Select **VMware kernel paravirtualization**.

Kernel support for VMI is included in some recent Linux distributions (Ubuntu 7.04 and later and SLES 10 SP2, for example), and can be compiled into other Linux distributions, typically by compiling the kernel with `CONFIG_PARAVIRT` and `CONFIG_VMI`. No Microsoft Windows operating systems support VMI. Check the *VMware Guest Operating System Installation Guide* to see which VMI operating systems are supported in Workstation. More information about VMI can be found in

Performance of VMware VMI (<http://www.vmware.com/resources/techresources/1038>) and the Paravirtualization API Version 2.5 (http://www.vmware.com/pdf/vmi_specs.pdf).

CPU Considerations in Guest Operating Systems

This section addresses CPU-related considerations within the guest operating system.

Guest Operating System UP Versus SMP HAL/Kernel

- Most guest operating systems can be configured to use either a *UP HAL/kernel* or an *SMP HAL/kernel*. The UP operating system versions are for single-processor systems. If used on a multiprocessor system, a UP operating system will recognize (and use) only one of the processors. The SMP versions, while required in order to fully utilize multiprocessor systems, may also be used on single-processor systems. Due to their extra synchronization code, however, SMP operating systems used on single-processor systems are slightly slower than UP operating systems.

Therefore, for best performance, a single-processor system should be configured to use a UP operating system. For more details on this topic, see VMware knowledge base article 1077, *High CPU Utilization of Inactive Virtual Machines*, at:

http://www.vmware.com/support/kb/enduser/std_adp.php?p_faqid=1077

- Most operating systems automatically select an appropriate HAL/kernel when they are first installed in a virtual machine: UP for single-processor virtual machines, SMP for dual-processor virtual machines. When a virtual machine with a UP HAL/kernel is reconfigured to have two processors, the typical behavior of the guest operating system is to automatically switch to an SMP HAL/kernel. If that virtual machine is later reconfigured to have a single processor, however, it typically does not automatically switch to a UP HAL/kernel.

NOTE Some newer versions of Windows, with appropriate BIOS and hardware support, may be able to seamlessly switch between UP and SMP HALs. More information about this topic can be found in the section titled *Virtual SMP Best Practices* within the white paper *Best Practices Using VMware Virtual SMP*, at: http://www.vmware.com/pdf/vsmp_best_practices.pdf

(this white paper primarily addresses VMware ESX Server, but the section referenced above also applies to VMware Workstation).

- For more information about working with SMP virtual machines, see: http://www.vmware.com/pdf/vsmp_best_practices.pdf

Guest Operating System Idle Loops

When a system is otherwise inactive, some operating systems might spin in their idle loops for some time before issuing halt instructions, whereas other operating systems might halt immediately. Typically, operating systems that spin in idle loops before halting are running SMP HALs/kernels.

In our experiments, we've found that some versions of Windows behave in this way when running SMP HALs, whereas other versions (or even the same version, but with a different service pack installed) halt quickly when they are idle.

The impact of this behavior may be reduced by adding (or modifying) the `monitor.idleLoopSpinUS=` variable in the applicable `.vmx` file, which overrides the idle loop handler's behavior on a per-virtual machine basis:

```
monitor.idleLoopSpinUS = "n"
```

In this case *n* is the number of microseconds the virtual machine is allowed to spin in its idle loop before it is halted by VMware Workstation. The default value is 2000, and the recommended range of settings is between 100 and 4000. Values outside this range are unlikely to result in resource-utilization improvements.

The `monitor.idleLoopSpinUS` variable has the following effects:

- A lower value means that the virtual machine is halted more quickly when idle (and thus uses fewer physical CPU cycles).
- A higher value means that the virtual machine is allowed to spin longer before being halted (and thus uses more physical CPU cycles). A higher value can also result in better virtual machine performance, however, because the CPU can wake up more quickly from spinning in its idle loop than from a halted state.

For further details please refer to VMware knowledge base article 1077, *High CPU Utilization of Inactive Virtual Machines*, at:

http://www.vmware.com/support/kb/enduser/std_adp.php?p_faaid=1077

and VMware knowledge base article 1730, *ESX Server Reports Increased CPU Utilization for Idle Microsoft Windows Server 2003 SP1 SMP Virtual Machines*, at:

http://www.vmware.com/support/kb/enduser/std_adp.php?p_faaid=1730

(the latter article specifically refers to VMware ESX Server 2.5.1, but the information applies equally to VMware Workstation).

Timer Interrupt Rates

Many operating systems keep time by counting timer interrupts. The timer interrupt rates vary between different operating systems and versions. For example:

- Unpatched 2.4 and earlier Linux kernels request timer interrupts at 100 Hz (100 interrupts per second).
- Older 2.6 Linux kernels and some 2.4 Linux kernels request interrupts at 1000 Hz.
- Newer 2.6 Linux kernels request interrupts at 250 Hz.
- Microsoft Windows operating system timer interrupt rates are specific to the version of Microsoft Windows and the Windows HAL that is installed. Windows systems typically use timer interrupt rates of 66 Hz or 100 Hz.
- Running applications that make use of the Microsoft Windows multimedia timer functionality can increase the timer interrupt rate. For example, some multimedia applications or Java applications increase the timer interrupt rate to 1000 Hz.

The total number of timer interrupts delivered to the virtual machine depends on a number of factors:

- Virtual machines running SMP HALs/kernels (even if they are running on a UP virtual machine) require more timer interrupts than those running UP HALs/kernels.
- The more virtual CPUs a virtual machine has, the more interrupts it requires.

Delivering many virtual timer interrupts negatively impacts guest performance and increases host CPU consumption. If you have a choice, use guest operating systems that require fewer timer interrupts. For example:

- If you have a UP virtual machine use a UP HAL/kernel.
- In RHEL 5.1 or later use the “`divider=10`” kernel boot parameter to reduce the timer interrupt rate to 100 Hz.

NOTE A bug in the RHEL 5.1 x86_64 kernel causes problems with the `divider` option. For RHEL 5.1 use the patch that fixes the issue at https://bugzilla.redhat.com/show_bug.cgi?id=305011. This bug is also fixed in RHEL 5.2. For more information see <http://rhn.redhat.com/errata/RHSA-2007-0993.html>.

- Use a VMI-enabled operating system and enable VMI for the virtual machine (see “[Running Paravirtualized Operating Systems](#)” on page 16).

For background information on this topic refer to *Timekeeping in Virtual Machines*, at:

http://www.vmware.com/pdf/vmware_timekeeping.pdf

Memory Considerations in Guest Operating Systems

The performance impact of page faults can be higher in virtual machines than in native systems. Therefore, for best performance, set the virtual machine's memory size to accommodate the guest's working set, thus avoiding excessive page faults.

To determine if page faults are an issue in a particular virtual machine:

In Windows:

Use the **Pages/sec** counter (which may be found in the **Memory Performance** object within Perfmon). For more information refer to:

<http://support.microsoft.com/kb/889654>

In Linux:

Run `stat` to display the `--swap--` counters including:

`si`: Amount of memory swapped in from disk (KB/s)

`so`: Amount of memory swapped out to disk (KB/s)

Ensure these two counters are as close to zero as possible.

For additional information about how page faults affect performance and how to measure them, see VMware knowledge base article 1687, *Excessive Page Faults Generated By Windows Applications May Impact the Performance of Virtual Machines*, at:

http://www.vmware.com/support/kb/enduser/std_adp.php?p_faaid=1687

Disk Considerations in Guest Operating Systems

- On Windows systems, the **Disk Properties** tab associated with each hard drive provides a check box enabling write caching on the disk and, in some cases, a check box enabling advanced performance on the disk. Selecting one or both of these features can improve the disk performance, especially in workloads that make heavy use of the disk.
- It is best to use virtual SCSI hard disks in the virtual machine. If you are using virtual IDE hard disks, however, and the guest operating system is Windows 95 OSR2 or later, you should make sure DMA access for the IDE hard disks is enabled.

For information about enabling DMA in Windows operating systems, see:

<http://support.microsoft.com/kb/258757>

<http://www.microsoft.com/whdc/device/storage/IDE-DMA.msp>

Defragment Disks

Defragment your disks on a regular basis for the best performance. This defragmentation must be done in stages, from inside the guest operating system out to the host operating system:

- 1 Defragment the virtual disks in the booted-up virtual machine using a guest operating system utility. For example, in a virtual machine running Windows XP, use the Windows XP Disk Defragmenter tool from within that virtual machine.

For the following reasons, VMware strongly recommends that you defragment the virtual disks this way before taking the first snapshot or creating the first linked clone:

- When you take a snapshot or create a linked clone, you permanently lose the ability to defragment inside the original disks. Instead, when you run a defragmenting program in the virtual machine after a snapshot, Workstation makes all its changes to the redo log, not to the original disks.
- When you defragment a heavily fragmented disk after a snapshot, every sector that is moved in the defragmentation process is recorded in the redo log, making the virtual machine redo log very large.

- 2 Use the VMware Workstation Defragmentation tool.

NOTE This step is required only for growable virtual disks. It is not required for preallocated virtual disks.

- With the virtual machine opened but powered off:
 - i Choose **VM > Settings**.
 - ii On the **Hardware** tab select **Hard Disk**.
 - iii Under the **Utilities** button, choose **Defragment**.
 - Alternatively, use the `vmware-vdiskmanager` command-line utility, which is found in the directory where VMware Workstation was installed (see the *VMware Virtual Disk Manager User's Guide* for more information about this command).
- 3 Defragment the host disks (for example, in a Windows Server 2003 Enterprise host, use the Windows Server 2003 defragment tool).

Performance Impact of Defragmenting Snapshots and Linked Clones

There might be a performance impact when you defragment a linked clone or a virtual machine with a snapshot. Exact performance degradation depends on:

- The fragmentation of the parent virtual machine disk when you created the snapshot or linked clone.
- The nature of the subsequent updates to the parent virtual machine disk.

The redo file itself can become fragmented. If your use of virtual machines is strongly performance oriented, you should avoid defragmenting—or using—snapshots and linked clones.

Networking Considerations in Guest Operating Systems

- You will obtain the best networking performance if your guest operating system uses one of the high-performance network drivers (`vmxnet` or `e1000`), instead of the default PCnet (`vlance`) driver.
- The default virtual network adapter emulated in a virtual machine is either an AMD PCnet device (`vlance`) or an Intel 82545EM device (`e1000`). However a special driver, called `vmxnet`, can be used with the `vlance` device. The `vmxnet` driver provides better performance than the default drivers for `vlance` or `e1000`, and should be used for optimal performance within any guest operating system for which it is available.

The `vmxnet` driver implements an idealized network interface that passes network traffic from the virtual machine to the physical cards with minimal overhead. Workstation supports `vmxnet` in most 32-bit guests and many 64-bit guests. The `vmxnet` driver is automatically installed when you install VMware Tools in a virtual machine.

In Workstation, “NIC morphing” (also called “flexible NIC”) automatically converts each `vlance` network device to a `vmxnet` network device for most guests in which `vmxnet` is supported. This only occurs if VMware Tools is installed within the guest (thus installing the `vmxnet` driver).

NOTE The network speeds reported by the network driver on the virtual machine do not necessarily reflect the actual speed of the underlying physical network interface card. For example, the `vlance` guest driver on a virtual machine reports a speed of 10Mbps, even if the physical card on the server is 100Mbps or 1Gbps, because the AMD PCnet cards that Workstation emulates are 10Mbps. However, Workstation is not limited to 10Mbps and transfers network packets as fast as the resources on the physical host machine allow.

- A new networking feature, TSO (TCP segmentation offload) is available in Workstation 6.5. The vmxnet device in virtual machines created with (or updated to) Workstation 6.5 hardware compatibility supports TSO. The vmxnet device in virtual machines created with Workstation 6 hardware compatibility (and not updated) does not support TSO.

Additionally, in order to use TSO with vmxnet, the network driver must be from Workstation 6.5 VMware Tools. This newer network driver is installed automatically when a Workstation 6.5 version of VMware Tools is installed. (The older network driver will work with the newer virtual network adapter, but this combination will not support TSO.)

No configuration action is required in order to use TSO in Workstation 6.5. It is used automatically when a guest is using the vmxnet driver (along with the newer virtual network adapter) or the e1000 driver.

Using TSO in a guest operating system can improve performance whether or not the underlying hardware supports TSO.

Software Version Considerations in Guest Operating Systems

- Make sure you are running general-availability (GA) releases of all operating systems, applications, and benchmarking software (rather than beta or debug versions) with all applicable patches and updates installed.
- Make sure you are using a guest operating system version that is supported by the VMware software you are using. This includes not just the type of operating system (for example, Linux), the variant (for example, Red Hat Enterprise Linux), and the version (for example, release 5), but even the specific kernel (for example, SMP).
- If available, consider using a paravirtualized kernel when running a Linux guest operating system. More information about paravirtualized kernels can be found in the *VMware Workstation User's Manual*.
- Make sure the most recent available version of the VMware Tools suite is installed on the virtual machine.

VMware Tools provides a suite of utilities and drivers. These utilities and drivers vary somewhat across guest operating systems, but typically include an optimized SVGA driver, a high-performance network driver, a mouse driver, the VMware Tools control panel, and support for such features as shared folders, defragmenting growable virtual disks, optional synchronization of the virtual machine's clock with the host computer's clock, VMware Tools scripts, and the ability to connect and disconnect devices while the virtual machine is running.

To install VMware Tools:

- a Make sure the virtual machine is powered on.
- b Choose **VM > Install VMware Tools**.

Guest Operating System Settings

- Check that any non-default system configuration settings are intentional.

Benchmarking VMware Workstation

Before running benchmark tests with VMmark Workstation, you should first tune the system for the best possible performance, as described in the first part of this guide.

This chapter picks up where the performance tuning chapter leaves off, discussing benchmark design and addressing those areas where tuning for a benchmark test might differ from tuning for performance.

NOTE Please check the relevant VMware product end-user license agreement (EULA) before publishing any benchmarking data regarding VMware products.

“Apples-to-Apples” Benchmarking Comparisons

It is important when doing performance comparisons to make sure that the configurations of the systems being compared are as similar as possible. Good performance comparisons only change one variable at a time. For example, if comparing the performance of VMware Workstation with another virtualization product, the hardware and software configurations should be equivalent, with the only difference being which virtualization product is used.

If there is a performance or resource cost associated with an optional feature or capability of a VMware product, that cost should be measured and reported separately from other benchmarking results.

For example, the use of growable disks (as opposed to the more traditional preallocated disks) within VMware Workstation can significantly reduce disk-space requirements. Growable disks can also have an impact on performance, and that performance impact should not be charged against virtualization. Instead, the increased disk-usage efficiency of growable disks is a potential added benefit of virtualization, but an optional one that should be weighed separately against its (usually) slight decrease in performance.

Benchmarking Design

This section provides tips and guidance on the design of benchmarking experiments.

General Benchmarking Methodology

- Before planning the testing, clearly define the parameters being measured and a metric with which to measure them (such as operations per second, jobs per hour, or average response time).
- If you are running a publicly-available benchmark (for example, SPEC*), make sure you follow all the guidelines associated with configuring, running, and reporting for that benchmark, both on the native system and within the virtual machine.
- If you are running a custom benchmark, make sure that it incorporates the well-understood principles of benchmarking: specific test purpose, a meaningful metric (such as time, operations per second, or bytes transferred), reproducibility, and so forth.
- Any report of the benchmark results should include enough details about the experimental setup to allow the reader to reproduce those results.

- When attempting to saturate and benchmark any specific system component (such as CPU, memory, or network), ensure that no other resource on the system is constrained in either the native or the virtual machine. Be aware that virtualization has overhead in terms of CPU, memory, etc., and provision for this overhead for components not being tested. For example, before making a native to virtual machine network-bandwidth comparison, make sure the processor load is not limiting the bandwidth achieved in either case.
- An ideal setup for workloads that involve network traffic is to use an external client (on a different physical system) to send network traffic to and receive network traffic from a virtual machine.

Timing Considerations When Benchmarking

- Timing numbers reported from within the virtual machine can be inaccurate, especially when the processor is overcommitted. If you report numbers from within the virtual machine, this fact should be noted in your results.
- One method of timing workloads is to ping an external machine and capture timestamps when that machine receives the pings. To use `ping` to time workloads, run `tcpdump` (in Linux) or `WinDUMP` (in Windows) on an external system. (The examples below show `tcpdump` in a C shell. `WinDUMP` works similarly.)

On the external system, run:

```
[timeserver]# tcpdump ip proto \\\icmp and host <vm_ip>
```

In this example, `<vm_ip>` is the IP address of the virtual machine under test. From within the virtual machine, before starting the benchmark and after the benchmark finishes execution, run:

```
[my_vm]$ ping -c 1 <external_system_ip>
```

In this case, `<external_system_ip>` is the IP address of the external system running `tcpdump` or `WinDUMP`. You can then determine the duration of the experiment by simply subtracting one timestamp from the other.

NOTE For more information about `tcpdump` and `WinDUMP`, see: <http://www.tcpdump.org/> and <http://www.winpcap.org/windump/>.

- In some cases your workload might not be able to use the above ping method, perhaps because the timing granularity is so small that the network latency of the pings is significant. It is also possible for the guest operating system to read the hardware timestamp counter (TSC) value as a pseudo performance counter using a VMware feature. The `rdpmc` command can be used to measure time from within the virtual machine by:
 - a Adding `monitor_control.pseudo_perfctr=1` to the virtual machine `.vmx` file.
 - b Issuing `rdpmc 0x10000` in the virtual machine.

More information about Linux guest timing issues can be found in VMware knowledge base article 1420, *Clock in a Linux Guest Runs More Slowly or Quickly Than Real Time*, at:

http://www.vmware.com/support/kb/enduser/std_adp.php?p_faqid=1420

- If the benchmark takes long enough, you can also use a stopwatch to measure time.
- More information about timekeeping in virtual machines can be found the timekeeping white paper available at:

http://www.vmware.com/pdf/vmware_timekeeping.pdf

Benchmarking Tools

This section lists some benchmarking tools that may help you compare the performance of various systems.

CPU-Related Benchmarks

- Passmark (available at <http://www.passmark.com/>)
- SPEC CPU2000 (available at <http://www.spec.org/cpu2000/>)

Memory-Related Benchmarks

- Passmark (available at <http://www.passmark.com/>)

Disk-Related Benchmarks

- Passmark (available at <http://www.passmark.com/>)
- IOMeter (available at <http://www.iometer.org/>)

Networking-Related Benchmarks

- Netperf (available at <http://www.netperf.org/netperf/NetperfPage.html>)
- SPECweb2005 (available at <http://www.spec.org/web2005/>)

Application Benchmarks

- SPECjbb2005 (available at <http://www.spec.org/jbb2005/>)
- SPECjAppServer2004 (available at <http://www.spec.org/jAppServer2004/>)
- SPECweb2005 (available at <http://www.spec.org/web2005/>)

Multiple-Virtual Machine Consolidation Benchmarks

- VMmark (available at <http://www.vmware.com/products/vmmark/>)

Discouraged Benchmarks

We don't recommend use of the following benchmarks, as our experience has shown that they can produce inconsistent results in virtual machines:

- Sisoft Sandra
- LMbench
- Unixbench

Performance Monitoring Tools

A number of tools are available for profiling Windows and Linux systems. Although these tools are useful for profiling native and host systems, their use within virtual machines should be avoided, due to unreliable results (the white paper *Timekeeping in VMware Virtual Machines*, at http://www.vmware.com/pdf/vmware_timekeeping.pdf, contains more information on this topic). These profiling tools include:

Windows: Perfmon

VMware Workstation incorporates a set of performance counters that work with Microsoft's Performance console so you can collect performance data from running virtual machines.

NOTE The Performance console is available only on Windows hosts. Therefore this performance information collected by VMware Workstation is not viewable when Workstation is running on a Linux host.

You can, however, monitor the performance of any virtual machines running on the Windows host, including those running Linux guest operating systems.

Perfmon can be used to profile all aspects of the system, including CPU, memory, disk, network, and so forth. To run Perfmon, choose **Start > Run**, then type `perfmon` in the **Open:** box.

In the Perfmon console, under **Console Root**, select **System Monitor**. Click the + sign to open the **Add Counters** dialog box. Under **Performance Object**, there are a number of objects of interest including **Memory**, **Network Interface**, **Physical Disk**, **Processor**, **System**, and **VMware**.

Once a counter is selected, click the **Explain** button to see detailed information regarding what the counter profiles, or click the **Add** button to display the information in the Performance console.

Further information about Perfmon can be found by choosing **Help** from within the application, or by visiting:

<http://www.microsoft.com/technet/prodtechnol/windowsserver2003/library/ServerHelp/8e9e5b72-13af-4525-9b60-52864b9d49b2.msp#>

Linux: vmstat

Invoke this tool by executing:

```
vmstat sample_interval num_times_to_sample
```

For example, `vmstat 1 10` samples every second and generates ten samples.

`vmstat` provides information regarding the CPU, memory, swap, disk, and system (interrupts, context switches).

Refer to the `vmstat` man page for more information.

Linux: sar

Invoke this tool by executing:

```
sar sample_interval num_times_to_sample
```

For example, `sar 1 10` samples every second and generates ten samples.

`sar` provides comprehensive information regarding the CPU, memory, swap, disk, system, processes, file system, and network. For the comprehensive list of counters monitored, run `sar -A`.

Refer to the `sar` man page for more information.

Hardware for Benchmarking

This section provides guidance about selecting and configuring hardware for use in benchmark tests of VMware Workstation.

General Hardware Considerations When Benchmarking

- In general for software performance comparisons it is best to run both tests on the same system. When this is not possible, use identical systems.
- Make sure to run all tests on hardware supported by the VMware software you are using.

CPU Considerations When Benchmarking

- For an “apples-to-apples” performance comparison between native and virtual machines, configure the same number of physical CPUs in the native system (while measuring the native performance) as you configured virtual CPUs in the virtual machine. This might require reducing the number of processors in your system when doing the native tests.

To do this in Windows, use the `/numproc` switch in the `boot.ini` file.

To do this in Linux, use the `maxcpus` variable in either the `grub.conf` or `lilo.conf` file (depending on your Linux version).

Memory Considerations When Benchmarking

- For an “apples-to-apples” performance comparison between native and virtual machines, configure the same amount of physical memory in the native system (while measuring the native performance) as you configured virtual memory in the virtual machine. This might require reducing the amount of memory in your system when doing the native tests.

To do this in Windows, use the `/maxmem` switch in the `boot.ini` file.

To do this in Linux, use the `mem` variable in either the `grub.conf` or `lilo.conf` file (depending on your Linux version).

- Though it might not be true for production environments, over-provisioning memory in experimental configurations may be a good idea, as over-provisioning can ensure that excessive page faults are avoided. This can allow other resources to be benchmarked without the need to worry about page faults.

Disks (SAN, NAS, and RAID) and Benchmarking

- If virtual disks are stored on a network storage device (such as SAN or NAS), make sure you are the only user connected to that device. If connecting through a switch (Fibre Channel or Ethernet), make sure your machine is the only one on the switch. Other traffic on a storage device or switch can affect your results.

Networking Considerations When Benchmarking

- Avoid cross-traffic noise over the network while conducting the experiments. Either use direct cables between the systems or use a private network switch. Use dedicated network interface cards on both machines for the connection.
- If possible, use similar network interface cards on systems under test so that they function well with each other. Using similar cards also helps to ensure that send and receive have similar performance. Ideally, you should use similar client and server machines as well, with similar system bus architectures and configurations. The differences between PCI, PCI-X, and PCIe, for example, can have a significant effect on networking performance.

Other Device Considerations When Benchmarking

- Remove or disable all devices that are not part of your experiment. These might include audio devices, optical drives (CD or DVD drives), floppy drives, USB ports and devices, network interface cards, and so on.

Configuring and Running VMware Workstation for Benchmarking

This section provides guidance about configuring and using VMware Workstation when running benchmarks.

Memory Trimming and Page Sharing in Workstation

- Because memory trimming and page sharing can increase disk I/O demands, it might be helpful to turn them off when benchmarking disk-intensive applications. In other situations, however, these two features might improve performance. (For more information about these functions, see [“Memory Trimming”](#) on page 13 and [“Page Sharing”](#) on page 13.)

Use Purpose-Built Virtual Machines When Benchmarking

It is good practice to run benchmarking tests on virtual machines that were created specifically for your performance tests (that is, purpose-built virtual machines) rather than reusing existing virtual machines. This is because existing virtual machines could have unnecessary applications or services installed that could impact performance, and could be configured with an incorrect HAL/kernel.

Creating a purpose-built virtual machine typically involves some or all of the following steps:

- 1 Create a fresh virtual machine.
- 2 Install an operating system.
- 3 Install any operating system upgrades or patches desired.
- 4 Install any applications or benchmarks that will be used.
- 5 Start and stop the machine to verify its correct functioning and to ensure that all hardware has been discovered and that all software installations have completed.
- 6 Check the log files for this virtual machine to make sure no errors or unexpected warnings are being reported.

Use this clean virtual machine, or clones of it, for all tests.

NOTE If you will be using clones, remember that *linked clones* do not perform as well as *full clones*. Therefore, when creating a clone, make sure to choose **From current state**, then **Create a full clone**.

Avoid running virtual machines from snapshots or linked clones.

Guest Operating Systems

This section provides guidance regarding the guest operating systems when running benchmark tests of VMware Workstation.

CPU Factors in Guest Operating Systems

This section addresses CPU factors in guest operating systems.

Guest Operating System UP Versus SMP HAL/Kernel

NOTE This section relies heavily on the information presented in [“Guest Operating System UP Versus SMP HAL/Kernel”](#) on page 17.

- When conducting SMP scaling experiments, it might be desirable to compare a system running an SMP operating system version on a single processor to a system running an SMP operating system version on two processors. If you keep the HAL/kernel the same in both systems you change only one variable (the number of processors). Note that this consideration applies to native, host, and virtual systems.
- When conducting experiments intended to help determine whether to run an application in a single-processor virtual machine or in a dual-processor virtual machine, it might be desirable to configure the single-processor machine with a UP HAL/kernel (since this provides the best single-processor performance) and the dual-processor machine with an SMP HAL/kernel (since this is the only choice available for a dual-processor system). This way, you configure each virtual machine to provide the best performance possible.
- If your benchmarks don't involve processor scaling, it might be best to stick with single-processor configurations (and UP HALs/kernels), thus avoiding unnecessary SMP overhead.
- If comparing a native system to a virtual machine, make sure the HAL/kernel types in both are the same (that is, either both UP or both SMP).
- When running a single-threaded benchmark, make sure you are using a UP HAL/kernel. Use of an SMP HAL/kernel in this case incurs SMP overhead without any benefits.
- When running a multithreaded benchmark with an SMP HAL/kernel, make sure there is enough parallelism to keep both CPUs busy.

32-bit Versus 64-bit CPUs

- Make sure you use similar types of operating system and application software (that is, 32-bit or 64-bit), both natively and within the virtual machine (for example, compare a 64-bit operating system running a 32-bit application natively to a 64-bit operating system running a 32-bit application within the virtual machine, not to a 64-bit operating system running a 64-bit application within the virtual machine).

Disk Considerations and Guest Operating Systems

- Defragment your disks before running benchmark tests for the best performance. This defragmentation must be done in stages, from the inside out, as described in [“Defragment Disks”](#) on page 19.

Other Guest Operating System Considerations

- Make sure you are running general-availability (GA) releases of all benchmarking software (rather than beta or debug versions).
- If you have tuned the native system (registry, swap space, and so forth), a similar tuning procedure should be performed on the virtual machine before comparing native to virtual machine performance. Settings on the native system might not be applicable within the virtual machine and it might therefore be necessary to repeat the full tuning process on the virtual machine.
- For workloads that include virtual machine startup (such as automated tests) make sure that the virtual machine is not detecting new hardware. If new hardware is detected, the **New Hardware Wizard** (on Windows) or kudzu (on Linux) might start up. In this case, complete the new hardware configuration process then save the virtual disk before performing the tests.
- Disable any programs or services not needed for your tests. Depending on your setup and environment, these might include screen savers, virus checkers, and so on. For further information, see [“Disable Unneeded Services”](#) on page 10.
- Direct the output from the application or benchmark to a log file instead of to the display. This avoids the added overhead associated with the virtualized display.

Glossary

- A** **AMD Virtualization (AMD-V)**
AMD's version of virtualization assist, included in some 64-bit AMD processors. See also Virtualization Assist.
- C** **Clone**
A copy of a virtual machine. See also Full Clone and Linked Clone.
- Console**
See VMware Virtual Machine Console.
- F** **Full Clone**
A copy of the original virtual machine that has no further dependence on the parent virtual machine. See also Linked Clone.
- Full Screen Mode**
A mode of operation in which a virtual machine's display occupies the host computer's entire screen. See also Normal (Windowed) Mode.
- G** **Growable Disk**
A type of virtual disk in which only as much host disk space as is needed is initially set aside, and the disk grows as the virtual machine uses the space. See also Preallocated Disk.
- Guest**
A virtual machine running within VMware Workstation. See also Virtual Machine.
- Guest Operating System**
An operating system that runs inside a virtual machine. See also Host Operating System.
- H** **Hardware Abstraction Layer (HAL)**
A layer between a computer's physical hardware and that computer's software designed to hide differences in the underlying hardware, thus allowing software to run on a range of different hardware architectures without being modified for each one. Windows uses different HALs depending, among other factors, on whether the underlying system has one CPU (Uniprocessor (UP) HAL) or multiple CPUs (Symmetric Multiprocessor (SMP) HAL). See also Kernel.
- Hardware Virtualization Assist**
See Virtualization Assist.

Host-Only Networking

A type of network connection between a virtual machine and the host. Under host-only networking, a virtual machine is connected to the host on a private network, which normally is not visible outside the host. Multiple virtual machines configured with host-only networking on the same host are on the same network. See also Network Address Translation.

Host Operating System

The operating system running directly on the host computer. VMware Workstation runs within the host operating system. See also Guest Operating System.

Host System (or Host)

A computer system on which VMware Workstation software is running.

Hyper-Threading

A processor architecture feature that allows a single processor to execute multiple independent threads simultaneously. Hyper-threading was added to Intel's Xeon and Pentium® 4 processors. Intel uses the term "package" to refer to the entire chip, and "logical processor" to refer to each hardware thread.

I Independent Virtual Disk

Independent virtual disks are not included in snapshots. Independent virtual disks can in turn be either Persistent or Nonpersistent.

K Kernel

The heart of an operating system. The kernel usually includes the functionality of a Hardware Abstraction Layer (HAL). Though applying to any operating system, the term is more often used in reference to Linux than to Windows.

L Linked Clone

A copy of the original virtual machine that must have access to the parent virtual machine's virtual disk(s). The linked clone stores changes to the virtual disk(s) in a set of files separate from the parent's virtual disk files. See also Full Clone.

N NAS

See Network Attached Storage.

Native Execution

Execution of an application directly on a physical server, as contrasted with running the application in a virtual machine.

Native System

A computer running a single operating system, and in which the applications run directly in that operating system.

Network Address Translation (NAT)

A type of network connection that allows you to connect your virtual machines to an external network when you have only one IP network address, and that address is used by the host computer. If you use NAT, your virtual machine does not have its own IP address on the external network. Instead, a separate private network is set up on the host computer. Your virtual machine gets an address on that private network from the VMware virtual DHCP server. The VMware NAT device passes network data between one or more virtual machines and the external network. It identifies incoming data packets intended for each virtual machine and sends them to the correct destination. See also Host-Only Networking.

Network-Attached Storage (NAS)

A storage system connected to a computer network. NAS systems are file-based, and often use TCP/IP over Ethernet (although there are numerous other variations). See also Storage Area Network.

Nonpersistent Disk

All disk writes issued by software running inside a virtual machine with a nonpersistent virtual disk appear to be written to disk, but are in fact discarded after the session is powered down. As a result, a disk in nonpersistent mode is not modified by activity in the virtual machine. See also Persistent Disk.

Normal (Windowed) Mode

A mode of operation in which a virtual machine's display resides within the VMware Workstation console screen. See also Full Screen Mode.

P **Pacifica**

A code name for AMD's version of virtualization assist, included in some 64-bit AMD processors. See AMD Virtualization.

Persistent Disk

All disk writes issued by software running inside a virtual machine are immediately and permanently written to a persistent virtual disk. As a result, a disk in persistent mode behaves like a conventional disk drive on a physical computer. See also Nonpersistent Disk.

Physical CPU

A processor within a physical machine. See also Virtual CPU.

Preallocated Disk

A type of virtual disk in which all the host disk space for the virtual machine is allocated at the time the virtual disk is created. See also Growable Disk.

S **SAN**

See Storage Area Network.

Secure Virtual Machine (SVM)

Another name for AMD's version of virtualization assist, included in some 64-bit AMD processors. See AMD Virtualization.

Service Console

The service console boots the systems and runs support, management, and administration applications.

Snapshot

A snapshot preserves the virtual machine just as it was when you took that snapshot — including the state of the data on all the virtual machine's disks and whether the virtual machine was powered on, powered off, or suspended. VMware Workstation lets you take a snapshot of a virtual machine at any time and revert to that snapshot at any time.

Storage Area Network (SAN)

A storage system connected to a dedicated network designed for storage attachment. SAN systems are usually block-based, and typically use the SCSI command set over a Fibre Channel network (though other command sets and network types exist as well). See also Network-Attached Storage.

Symmetric Multiprocessor (SMP)

A multiprocessor architecture in which two or more processors are connected to a single pool of shared memory. See also Uniprocessor (UP).

T **Template**

A virtual machine that cannot be deleted or added to a team. Setting a virtual machine as a template protects any linked clones or snapshots that depend on the template from being disabled inadvertently.

Thrashing

A situation that occurs when virtual or physical memory is not large enough to hold the full working set of a workload. This mismatch can cause frequent reading from and writing to a paging file, typically located on a hard drive, which can in turn severely impact performance.

TSO (TCP Segmentation Offload)

A feature of some NICs that offloads the packetization of data from the CPU to the NIC.

U Uniprocessor (UP)

A single-processor architecture. See also Symmetric Multiprocessor (SMP).

V Vanderpool

A code name for Intel's version of virtualization assist, included in some 64-bit Intel processors. See Virtualization Technology.

Virtual CPU

A processor within a virtual machine. VMware Workstation currently supports up to two virtual CPUs per virtual machine.

Virtual Disk

A virtual disk is a file or set of files that appears as a physical disk drive to a guest operating system. These files can be on the host machine or on a remote file system. When you configure a virtual machine with a virtual disk, you can install a new operating system into the disk file without the need to repartition a physical disk or reboot the host.

Virtual Machine

A virtualized x86 PC environment in which a guest operating system and associated application software can run. Multiple virtual machines can operate on the same host system concurrently.

Virtual SMP™

A VMware proprietary technology that supports multiple virtual CPUs in a single virtual machine.

Virtualization Assist

A general term for technology included in some 64-bit processors from AMD and Intel that can allow 64-bit operating systems to be run in virtual machines (where supported by VMware Workstation). More information is available in VMware knowledge base article 1901. See also AMD Virtualization and Virtualization Technology.

Virtualization Overhead

The cost difference between running an application within a virtual machine and running the same application natively. Since running in a virtual machine requires an extra layer of software, there is by necessity an associated cost. This cost might be additional resource utilization or decreased performance.

Virtualization Technology (VT)

Intel's version of virtualization assist, included in some 64-bit Intel processors. See also Virtualization Assist.

VMware Tools

A suite of utilities and drivers that enhances the performance and functionality of your guest operating system. Key features of VMware Tools include some or all of the following, depending on your guest operating system: an SVGA driver, a mouse driver, the VMware Tools control panel, and support for such features as shared folders, shrinking virtual disks, time synchronization with the host, VMware Tools scripts, and connecting and disconnecting devices while the virtual machine is running.

VMware Virtual Machine Console

An interface that provides access to one or more virtual machines running on the host. You can view the virtual machines' displays to run programs within them or modify guest operating system settings. In addition, you can change the virtual machines' configurations, install a guest operating system, or choose to run a virtual machine in full screen mode.

