

Performance Best Practices for VMware Workstation

VMware Workstation 7.0

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see <http://www.vmware.com/support/pubs>.

EN-000294-00

vmware[®]

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

Copyright © 2007–2009 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>.

VMware is a registered trademark or trademark 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	5
Document Feedback	5
Technical Support and Education Resources	5
Online and Telephone Support	5
Support Offerings	5
VMware Professional Services	6
1 Hardware for VMware Workstation	7
CPUs for VMware Workstation	7
Hyperthreading	7
Hardware-Assisted Virtualization	7
Hardware-Assisted CPU Virtualization (Intel VT-x and AMD AMD-V)	8
Hardware-Assisted MMU Virtualization (Intel EPT and AMD RVI)	8
Memory for VMware Workstation	8
Storage for VMware Workstation (General)	8
Storage for VMware Workstation (SAN, NAS, and RAID)	9
Networking for VMware Workstation	9
Hardware BIOS Settings	10
2 Host Operating System Configuration	11
Memory and the Host Operating System	11
Storage and the Host Operating System	11
Networking and the Host Operating System	12
Services and Settings in the Host Operating System	12
Disable Unneeded Services	12
In Windows	12
In Linux	13
Avoid Non-Default Configuration Settings	13
3 VMware Workstation and Virtual Machines	15
Workstation General Considerations	15
CPU Considerations in Workstation and Virtual Machines	15
CPU Resources	15
Configuring Process Scheduling on Windows Hosts	17
Configuring Workstation for Hardware-Assisted Virtualization	17
Memory Resources in Workstation and Virtual Machines	18
Reserving Memory for Workstation	18
Specifying the Amount of Memory That Can Be Swapped to Disk	18
Memory Allocated to Virtual Machines	19
Memory Overcommit Techniques	19
Page Sharing	20
Memory Trimming	21
Hardware-Assisted MMU Virtualization	21
Storage Considerations in Workstation and Virtual Machines	21

Networking Considerations in Workstation and Virtual Machines	22
Debug and Logging Modes in Workstation	22
Turn off Debug Mode in Workstation	22
Logging in Workstation	23
Other Workstation and Virtual Machine Considerations	23
4 Guest Operating Systems	25
Guest Operating System General Considerations	25
Running Paravirtualized Operating Systems	26
Measuring Performance in Virtual Machines	27
CPU Considerations in Guest Operating Systems	27
Guest Operating System UP Versus SMP HAL/Kernel	27
Guest Operating System Idle Loops	28
Timer Interrupt Rates	28
Memory Considerations in Guest Operating Systems	29
In Windows	29
In Linux	29
Storage Considerations in Guest Operating Systems	30
Defragment Disks	30
Performance Impact of Defragmenting Snapshots and Linked Clones	31
Networking Considerations in Guest Operating Systems	31
Software Version Considerations in Guest Operating Systems	32
Guest Operating System Settings	32
Glossary	33
Index	39

About This Book

This book provides guidance in obtaining the best possible performance using VMware® Workstation 7.0, both in a production environment and when running benchmarks. Though this book focuses on Workstation 7.0, most of the guidelines presented here also apply to prior versions of Workstation and to VMware Server and VMware Player.

We address configuration of the host, the Workstation software, and the operating systems and applications in the individual virtual machines, including areas in which the best-performing configurations of Workstation virtual machines vary slightly from the configurations of native machines.

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

Definitions for terms in italic font can be found in the “[Glossary](#)” on page 33.

Intended Audience

This book is intended for anyone who wants to maximize the performance of 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

To 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.

Support Offerings

To 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/>.

Hardware for VMware Workstation

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

This chapter includes the following sections:

- [“CPUs for VMware Workstation”](#) on page 7.
- [“Memory for VMware Workstation”](#) on page 8.
- [“Storage for VMware Workstation \(General\)”](#) on page 8.
- [“Storage for VMware Workstation \(SAN, NAS, and RAID\)”](#) on page 9.
- [“Networking for VMware Workstation”](#) on page 9.
- [“Hardware BIOS Settings”](#) on page 10.

CPUs for VMware Workstation

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

Hyperthreading

Hyperthreading technology (recent versions of which are called symmetric multithreading, or SMT) allows a single physical processor core to behave like two logical processors, essentially allowing two independent threads to run simultaneously. Unlike having twice as many processor cores, which can roughly double performance, hyperthreading can provide anywhere from a slight to a significant increase in system performance by keeping the processor pipeline busier.

If the hardware and BIOS support hyperthreading, most host operating systems automatically make use of it. For the best performance, enable hyperthreading as follows:

- 1 Ensure that your system supports hyperthreading technology.
It is not enough that the processors support hyperthreading — the BIOS must support it as well. Consult your system documentation to see if the BIOS includes support for hyperthreading.
- 2 Enable hyperthreading in the system BIOS.
Some manufacturers label this option **Logical Processor**, others label it **Enable Hyper-threading**.

Hardware-Assisted Virtualization

Many recent processors from both Intel and AMD include hardware features to assist virtualization. These features were released in two generations: the first generation introduced CPU virtualization, the second generation included CPU virtualization and added memory management unit (MMU) virtualization. For the best performance, make sure your system uses processors with second-generation hardware-assist features.

Hardware-Assisted CPU Virtualization (Intel VT-x and AMD AMD-V)

The first generation of hardware virtualization assistance, VT-x from Intel and AMD-V from AMD, became available in 2006. These technologies automatically trap sensitive calls, eliminating the overhead required to do so in software. This allows the use of a hardware virtualization (HV) virtual machine monitor (VMM) as opposed to a binary translation (BT) VMM.

Hardware-Assisted MMU Virtualization (Intel EPT and AMD RVI)

Some recent processors also include a new feature that addresses the overheads due to memory management unit (MMU) virtualization by providing hardware support to virtualize the MMU. Workstation 7.0 supports this feature both in AMD processors, where it is called rapid virtualization indexing (RVI) or nested page tables (NPT), and in Intel processors, where it is called extended page tables (EPT).

Without hardware-assisted MMU virtualization, the guest operating system maintains guest virtual memory to guest physical memory address mappings in guest page tables, while 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, because the hardware translation lookaside buffer (TLB) will cache direct guest virtual memory to host physical memory address translations read from the shadow page tables. However, extra work is required to maintain the shadow page tables.

Hardware-assisted MMU 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 virtualization, see [“Configuring Workstation for Hardware-Assisted Virtualization”](#) on page 17.

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.

Because memory overcommitment can allow virtual machines to use less physical memory than the sum of their virtual memory allocations, the last bullet above is not simply the sum of the memory allocated to all running virtual machines. For more information about memory overcommitment, see [“Memory Overcommit Techniques”](#) on page 19.

Storage 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 Workstation has to use a nearly-full host hard drive when working with snapshots or writing to redo logs.

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

Storage performance issues are often the result of storage hardware misconfiguration. 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: 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

This section addresses networking considerations when selecting and configuring hardware.

- 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.
- Do not 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.
- Make sure the NICs are configured to use autonegotiation to set speed and duplex settings and are configured for full-duplex mode.
- For the best networking performance, VMware recommends the use of network adapters that support the following hardware features:
 - Checksum offload
 - TCP segmentation offload (TSO)
 - Ability to handle high-memory DMA (that is, 64-bit DMA addresses)
 - Ability to handle multiple Scatter Gather elements per Tx frame
 - Jumbo frames (JF)

Hardware BIOS Settings

The default hardware BIOS settings might not always be the best choice for optimal performance. This section lists some of the BIOS settings you might want to check.

NOTE Enhanced Intel SpeedStep, Enhanced AMD PowerNow!, and other CPU power management technologies can save power when a host is not fully utilized. However, because these power-saving technologies can reduce performance in some situations, you should consider disabling them when performance considerations outweigh power considerations.

NOTE Because of the large number of different hardware models and configurations, any list of BIOS options is always likely to be incomplete.

- Make sure you are running the latest version of the BIOS available for your system.

NOTE After updating the BIOS, you might have to make sure your BIOS settings are still as you wish.

- Make sure the BIOS is set to enable all populated sockets and enable all cores in each socket.
- Enable Turbo Mode if your processor supports it.
- Enable 64-bit mode if your processor supports it.
- Make sure hyperthreading is enabled in the BIOS.
- Some NUMA-capable systems provide an option to disable NUMA by enabling node interleaving. In most cases you get the best performance by disabling node interleaving, thus enabling NUMA.
- Make sure any hardware-assisted virtualization features (VT-x, AMD-V, EPT, RVI) are enabled in the BIOS.

NOTE After these changes are made some systems might need a complete power down before the changes take effect. See <http://communities.vmware.com/docs/DOC-8978> for details.

- Disable C1E halt state in the BIOS. See note above regarding performance considerations versus power considerations.
- Disable any other power-saving mode in the BIOS. See note above regarding performance considerations versus power considerations.

Host Operating System Configuration

This chapter provides guidance regarding configuring the host operating system used with Workstation and other settings that are made from within the host operating system.

This chapter includes the following sections:

- “[Memory and the Host Operating System](#)” on page 11.
- “[Storage and the Host Operating System](#)” on page 11.
- “[Networking and the Host Operating System](#)” on page 12.
- “[Services and Settings in the Host Operating System](#)” on page 12.

Memory and the Host Operating System

Make sure memory is sized 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 29.

Storage 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, a check box for enabling advanced performance on the disk. Selecting one or both of these check boxes within the host operating system can improve the host’s disk performance in general. Selecting them for the host disks containing Workstation virtual disk files can improve virtual machine disk performance in particular, especially when that virtual machine is making heavy use of the disk.



CAUTION 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).
- Do not 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, 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 nondefault system configuration settings are intentional.

VMware Workstation and Virtual Machines

3

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

This chapter includes the following sections:

- [“Workstation General Considerations”](#) on page 15.
- [“CPU Considerations in Workstation and Virtual Machines”](#) on page 15.
- [“Memory Resources in Workstation and Virtual Machines”](#) on page 18.
- [“Storage Considerations in Workstation and Virtual Machines”](#) on page 21.
- [“Networking Considerations in Workstation and Virtual Machines”](#) on page 22.
- [“Debug and Logging Modes in Workstation”](#) on page 22.
- [“Other Workstation and Virtual Machine Considerations”](#) on page 23.

Workstation General Considerations

Allocate only as much virtual hardware as required for each virtual machine. Provisioning a virtual machine with more resources than it requires can, in some cases, reduce the performance of that virtual machine and other virtual machines sharing the same host.

Workstation 6.5 introduced virtual hardware version 7. By creating virtual machines using this hardware version, or upgrading existing virtual machines to this version, a number of additional capabilities become available. This hardware version is not compatible with previous versions of Workstation, however, and thus virtual machines running on hardware version 7 will be constrained to run only on Workstation 6.5 or later hosts or on ESX 4.0 or later hosts.

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.

CPU virtualization adds varying amounts of overhead depending on the percentage of the virtual machine's workload that can be executed on the physical processor as is and the cost of virtualizing the remaining workload:

- For many workloads, CPU virtualization adds only a very small amount of overhead, resulting in performance essentially comparable to native.
- Many workloads to which CPU virtualization does add overhead are not CPU-bound. That is, most of their time is spent waiting for external events such as user interaction, device input, or data retrieval, rather than executing instructions. Because otherwise-unused CPU cycles are available to absorb the virtualization overhead, these workloads typically have throughput similar to native, but potentially with a slight increase in latency.
- For a small percentage of workloads, for which CPU virtualization adds overhead and which are CPU-bound, there might be a noticeable degradation in both throughput and latency.

The rest of this subsection lists practices and configurations recommended by VMware for optimal CPU performance.

- When configuring virtual machines, the total CPU resources needed by the virtual machines running on the system should not exceed the CPU capacity of the host. If the host CPU capacity is overloaded, the performance of individual virtual machines may degrade.
- Periodically monitor the CPU usage of the host. This can be done in the host operating system by using PerfMon (in Windows) or `top` (in Linux).
- Configure only as many virtual CPUs (vCPUs) as your workload can use. For example, do not use virtual SMP if your application is single-threaded and will not benefit from the additional vCPUs, and if you do use virtual SMP, avoid configurations that leave some of the vCPUs idle the majority of the time.

Even if some vCPUs are not used, configuring virtual machines with them still imposes some small resource requirements on Workstation:

- Unused vCPUs still consume timer interrupts.
- Maintaining a consistent memory view among multiple vCPUs consumes resources.
- Some older guest operating systems execute idle loops on unused vCPUs, thereby consuming resources that might otherwise be available for other virtual machines or for the host operating system.
- The guest scheduler might migrate a single-threaded workload amongst multiple vCPUs, thereby losing cache locality.

Though small, these resource requirements translate to real CPU consumption from the point of view of the host operating system.

- Although some recent operating systems (including Windows Vista, Windows Server 2008, and Windows 7) use the same HAL (hardware abstraction layer) or kernel for both UP and SMP installations, many operating systems can be configured to use either a UP HAL/kernel or an SMP HAL/kernel. To obtain the best performance on a single-vCPU virtual machine running an operating system that offers both UP and SMP HALs/kernels, configure the operating system with a UP HAL or kernel.

The UP operating system versions are for single-processor systems. If used on a multiprocessor system, a UP operating system version 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 system versions used on single-processor systems are slightly slower than UP operating system versions used on the same systems.

NOTE When changing an existing virtual machine running Windows from multiprocessor to single-processor the HAL usually remains SMP.

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.

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-Assisted Virtualization

For a description of hardware-assisted virtualization, see [“Hardware-Assisted Virtualization”](#) on page 7.

Hardware-assisted 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 7.0 will enable or disable the use of hardware-assisted 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-assisted virtualization for a specific virtual machine

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

Memory Resources in Workstation and Virtual Machines

This section addresses memory 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 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's ability 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 be powered on 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 each virtual machine's memory that the host operating system can swap to disk. To change this setting, choose **Edit > Preferences**, select the **Memory** tab, and under **Additional memory** 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 **Memory** panel. This setting imposes the tightest restrictions on the quantity and memory size of virtual machines that can run simultaneously. Because Workstation tries to run the virtual machines entirely in RAM, they have the best possible performance.
- **Allow some virtual machine memory to be swapped** – Allow enough virtual machines to be powered on that the host operating system might need to swap a moderate amount of virtual machine memory to disk. 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** – Allow enough virtual machines to be powered on that the host operating system might need to swap most virtual machine memory to disk. This setting allows you to run even more virtual machines with even more memory than the intermediate setting does. This option can result in lower performance than the other options.

NOTE The setting described in this section affects only whether or not a virtual machine is allowed to start, not what happens after it starts (with one exception: the setting can also affect the level of memory trimming applied). After a virtual machine starts, other factors, such as applications running in the host operating system, can change. In such situations, even if you selected the **Fit all virtual machine memory into reserved host RAM** option, virtual machine memory swapping might still occur.

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, thus minimizing swapping within the guest operating system.

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 more memory to one virtual machine can reduce the amount of memory available to other virtual machines or to the host operating system. In some cases, this can reduce overall system performance.
- In situations where memory pressure reduces the performance of virtual machines, consider freeing memory by closing other applications running in the host operating system, especially applications that lock memory.
- When possible, configure 32-bit Linux virtual machines with no more than 896MB of memory. 32-bit Linux kernels use different techniques to map memory on systems with more than 896MB. These techniques impose additional overhead on the virtual machine monitor and can result in slightly reduced performance.

Memory Overcommit Techniques

Workstation 7.0 uses three memory management mechanisms, page sharing, ballooning, and swapping, to dynamically reduce the amount of machine physical memory required for each virtual machine.

- **Page Sharing.** Workstation uses a proprietary technique to transparently and securely share memory pages between virtual machines, thus eliminating redundant copies of memory pages. Page sharing is used by default regardless of the memory demands on the host system. See “Page Sharing” on page 20 for additional detail.
- **Ballooning.** If the virtual machine’s memory usage approaches its memory target, Workstation uses ballooning to reduce that virtual machine’s memory demands. Using a VMware-supplied `vmmemctl` module installed in the guest operating system as part of VMware Tools suite, Workstation can cause the guest to relinquish the memory pages it considers least valuable. Ballooning provides performance closely matching that of a native system under similar memory constraints. To use ballooning, the guest operating system must be configured with sufficient swap space.
- **Swapping.** If ballooning fails to sufficiently limit a virtual machine’s memory usage, Workstation makes memory pages available for the host operating system to swap. Because the pages Workstation releases are likely to be active, this can cause virtual machine performance to degrade significantly.

While Workstation allows significant memory overcommitment, usually with little or no impact on performance, you should avoid overcommitting memory to the point that it results in heavy memory reclamation.

If the memory savings provided by transparent page sharing reaches its limit and the total memory demands exceed the amount of machine memory in the system, Workstation must use additional memory reclamation techniques. As described above, Workstation next uses ballooning to make memory available to virtual machines that need it, typically with little performance impact. If still more reclamation is needed, Workstation uses host-level swapping, which can result in noticeable performance degradation.

If you suspect that memory overcommitment is affecting the performance of a virtual machine you can do the following:

- 1 Check for swap activity within that virtual machine.
This can indicate that ballooning might be starting to impact performance (though swap activity can also be related to other issues entirely within the guest).
- 2 Check for swap activity within the host operating system.
Swap activity in the host can be caused by memory demands from Workstation and from the virtual machines running within it, but it can also be caused by memory demands from other applications running on the host.

NOTE Methods to check for swap activity in Windows or Linux can be found in [“Memory Considerations in Guest Operating Systems”](#) on page 29.

A lack of swap activity in both the guest virtual machine and the host indicates that memory overcommitment is not affecting performance. In addition, because some swapping is normal in any operating system, a small amount of swap activity in the host and guest is not an indication of problems.

However significant swap activity in either the host or the guest is an indication that memory overcommitment might be affecting performance. To alleviate this, reduce memory pressure by shutting down applications in either the host or the guest.

Page Sharing

As described in [“Memory Overcommit Techniques,”](#) 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"
```

Memory Trimming

In addition to the memory overcommit techniques described in [“Memory Overcommit Techniques”](#) on page 19, Workstation also uses memory trimming to slowly deallocate lightly-used virtual machine memory for the host to reallocate, thus reducing Workstation’s overall host memory usage. Memory trimming can be helpful in low-memory situations and, because the default rate of memory trimming is very low, usually has little impact on performance. However, memory trimming can very slightly reduce disk performance in virtual machines. To disable memory trimming, use either one of the following methods:

- Add “MemTrimRate=0” to the .vmx file for the virtual machine.
(MemTrimRate is the maximum number of memory pages released per second for an idle virtual machine. The default setting is 30.)
- 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**.

NOTE The memory allocation settings described in [“Specifying the Amount of Memory That Can Be Swapped to Disk”](#) on page 18 also affect memory trimming. In particular, selecting **Fit all virtual machine memory into reserved host RAM** will disable memory trimming for all virtual machines.

Hardware-Assisted MMU Virtualization

Hardware-assisted MMU virtualization is a technique that uses hardware features to help virtualize the CPU’s memory management unit (MMU). For a description of this feature, see [“Hardware-Assisted MMU Virtualization \(Intel EPT and AMD RVI\)”](#) on page 8; for information about configuring the way Workstation uses this feature, see [“Configuring Workstation for Hardware-Assisted Virtualization”](#) on page 17.

Storage 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.
- 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.

- For the best performance, 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.
- *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.

To configure a virtual machine for independent persistent disks

- 1 With the virtual machine opened but powered off, choose **VM > Settings**.
- 2 On the **Hardware** tab select **Hard Disk**.

- 3 Click the **Advanced** button in the lower right.
- 4 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

- 1 Choose **VM > Snapshot > Snapshot Manager**.
- 2 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.
- Though the network performance of network address translation (NAT) network connections is comparable to that of bridged-mode network connections, the NAT connections typically have slightly higher CPU utilization. If CPU resources are an issue, consider using bridged mode.
- To obtain the best performance and lowest CPU overhead on a network connection between two virtual machines that reside on the same system, consider using virtual machine communications interface sockets (VMCI Sockets). This is an API that facilitates high throughput communication between virtual machines, avoiding unnecessary CPU and network overhead.

NOTE At present, no off-the-shelf software uses VMCI Sockets. Thus using this capability will require custom software or modification of open-source packages.

Debug and Logging Modes in Workstation

This section addresses the debugging and logging modes in Workstation.

Turn off Debug Mode in Workstation

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

- Workstation has logging enabled by default. Although this generates disk I/O, the impact is extremely small, and VMware recommends 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_faqid=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 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

- Choose **VM > Settings**.
- On the **Options** select **General**.
- Look in the **Version** window.

Note that to change the selection you must first power off the virtual machine.

- Check that any non-default 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 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 Workstation does not connect to your host optical drive).

To make this change

- Choose **VM > Settings**.
- On the **Hardware** tab select **CD/DVD**.
- 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 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 chapter provides guidance regarding the guest operating systems running in virtual machines.

This chapter includes the following sections:

- [“Guest Operating System General Considerations”](#) on page 25.
- [“CPU Considerations in Guest Operating Systems”](#) on page 27.
- [“Memory Considerations in Guest Operating Systems”](#) on page 29.
- [“Storage Considerations in Guest Operating Systems”](#) on page 30.
- [“Networking Considerations in Guest Operating Systems”](#) on page 31.
- [“Software Version Considerations in Guest Operating Systems”](#) on page 32.
- [“Guest Operating System Settings”](#) on page 32.

Guest Operating System General Considerations

This section addresses general guest operating system considerations.

- Use guest operating systems that are supported by Workstation. See the *Guest Operating System Installation Guide* for a list.

NOTE VMware Tools might not be available for unsupported guest operating systems.

- Install the latest version of VMware Tools in the guest operating system. Make sure to update VMware Tools after each Workstation upgrade.

Installing VMware Tools in Windows guests updates the BusLogic SCSI driver included with the guest operating system to the VMware-supplied driver. The VMware driver has optimizations that guest-supplied Windows drivers do not.

VMware Tools also includes the balloon driver used for memory reclamation in Workstation. Ballooning (described in [“Memory Overcommit Techniques”](#) on page 19) will not work if VMware Tools is not installed.

- Disable screen savers and Window animations in virtual machines. On Linux, if using an X server is not required, disable it. Screen savers, animations, and X servers all consume extra physical CPU resources, potentially affecting consolidation ratios and the performance of other virtual machines.

- Schedule backups and antivirus programs in virtual machines to run at off-peak hours and avoid scheduling them to run simultaneously in multiple virtual machines on the same host. In general, evenly distribute CPU usage, not just across CPUs, but also across time. For workloads such as backups and antivirus where the load is predictable, this is achieved by scheduling the jobs appropriately.
- For the most accurate timekeeping, consider configuring your guest operating system to use NTP, Windows Time Service, or another timekeeping utility suitable for your operating system. The time-synchronization option in VMware Tools can be used instead, but it is not designed for the same level of accuracy as these other tools and does not adjust the guest time when it is ahead of the host time. We recommend, however, that within any particular virtual machine you use either the VMware Tools time-synchronization option, or another timekeeping utility, but not both.

Running Paravirtualized Operating Systems

Workstation 7.0 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

- 1 Choose **VM > Settings**.
- 2 On the **Hardware** tab select **Processors**.
- 3 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).

For best performance, consider the following regarding enabling VMI support:

- If running 32-bit Linux guest operating systems that include kernel support for VMI on hardware that does not support hardware-assisted MMU virtualization (EPT or RVI), enabling VMI will improve performance.
- VMI-enabled virtual machine always use Binary Translation (BT) and shadow page tables, even on systems that support hardware-assisted MMU virtualization. Because hardware-assisted MMU virtualization almost always provides more of a performance increase than VMI, VMware recommends disabling VMI for virtual machines running on hardware that supports hardware-assisted MMU virtualization. No kernel change is required in the guest, as the VMI kernel can run in a non-VMI enabled virtual machine.

Measuring Performance in Virtual Machines

Be careful when measuring performance from within virtual machines.

- Timing numbers measured from within virtual machines can be inaccurate, especially when the processor is overcommitted.

NOTE One possible approach to this issue is to use a guest operating system that has good timekeeping behavior when run in a virtual machine, such as a guest that uses the `NO_HZ` kernel configuration option (sometimes called “tickless timer”) or the VMI paravirtual timer. More information about this topic can be found in *Timekeeping in VMware Virtual Machines* (http://www.vmware.com/pdf/vmware_timekeeping.pdf).

- Measuring performance from within virtual machines can fail to take into account resources used by the host operating system for tasks it has offloaded from the guest operating system and resources consumed by virtualization overhead.

Measuring resource utilization using tools at the host level, such as PerfMon in Windows or `top` in Linux can avoid these problems.

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, can 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, might 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 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.

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 might 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 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, 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 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_faqid=1730

The latter article specifically refers to VMware ESX Server 2.5.1, but the information applies equally to 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.
- The most recent 2.6 Linux kernels introduce the `NO_HZ` kernel configuration option (sometimes called “tickless timer”) that uses a variable timer interrupt rate.
- 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 between 66 Hz and 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 vCPUs 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>.

- Kernels with tickless-timer support (NO_HZ kernels) do not schedule periodic timers to maintain system time. As a result, these kernels reduce the overall average rate of virtual timer interrupts, thus improving system performance and scalability on hosts running large numbers of virtual machines.
- Use a VMI-enabled operating system and enable VMI for the virtual machine (see “[Running Paravirtualized Operating Systems](#)” on page 26).

More information about this topic can be found in *Timekeeping in VMware Virtual Machines* (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 (as described in “[Memory Allocated to Virtual Machines](#)” on page 19), 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 (KBps)

so: Amount of memory swapped out to disk (KBps)

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_faqid=1687.

Storage Considerations in Guest Operating Systems

- 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, a check box for enabling advanced performance on the disk. Selecting one or both of these features within the guest operating system can improve the guest's disk performance, especially in workloads that make heavy use of the disk.



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

- 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>
- If you choose to use the BusLogic virtual SCSI adapter and are using a Windows guest operating system, you should use the custom BusLogic driver included in the VMware Tools package.
- The depth of the queue of outstanding commands in the guest operating system SCSI driver can significantly impact disk performance. A queue depth that is too small, for example, limits the disk bandwidth that can be pushed through the virtual machine. See the driver-specific documentation for more information on how to adjust these settings.
- Large I/O requests issued by applications in the guest can be split by the guest storage driver. Changing the guest registry settings to issue larger block sizes can eliminate this splitting, thus enhancing performance. For additional information see KB article 9645697.
- Make sure that file system partitions within the guest are aligned.

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.

To defragment disks

- 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 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 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 (vlnace) driver.

The default virtual network adapter emulated in a virtual machine is either an AMD PCnet device (vlnace) or an Intel 82545EM device (E1000). However VMware also offers a paravirtualized network driver, called VMXNET, that can be used with the vlnace device. The VMXNET driver provides better performance than the default driver for vlnace and should be used for optimal performance within any guest operating system in which Workstation selects vlnace as the default device and for which the VMXNET driver 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. VMXNET drivers are available for most guest operating systems supported by Workstation. 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 vlnace network device to a VMXNET network device for most guests in which VMXNET is supported. This only occurs, however, 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 vlnace 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.

- In Workstation, TSO is supported in guests only when they are using the E1000 or VMXNET drivers. TSO can improve performance even if the underlying hardware does not support TSO.

- Jumbo frames are not supported in Workstation, even though some driver user interfaces might offer the option to enable them.

NOTE This means that when moving a virtual machine from ESX to Workstation, you should make sure the virtual machine is not configured to use jumbo frames.

- In some cases, low receive throughput in a virtual machine can be caused by insufficient receive buffers in the receiver network driver. If the receive ring in the guest operating system's network driver overflows, packets will be dropped, degrading network throughput. A possible workaround is to increase the number of receive buffers, though this may increase the host physical CPU workload.

For VMXNET in Workstation, the default number of receive and transmit buffers are 150 and 256, respectively, with the maximum possible for both being 512. You can alter these settings by changing the buffer size defaults in the `.vmx` (configuration) files for the affected virtual machines. For additional information see KB article 1428.

For E1000, the default number of receive and transmit buffers are controlled by the guest driver. For Linux, the values for E1000 can be changed from within the guest by using `ethtool`. In Windows, the values for E1000 can be changed in the device properties window.

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 nondefault system configuration settings are intentional.

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 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. Workstation runs within the host operating system. See also Guest Operating System.

host system (or host)

A computer system on which 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 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. 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. 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 machine communication interface (VMCI) sockets

First introduced in VMware Workstation 6.5 and VMware Server 2.0, VMCI Sockets is a convenience API for the VMCI device. With minimal code modifications, VMCI Sockets can provide fast communication between a guest virtual machine and its host and between guest virtual machines.

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. Because 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.

Index

Numerics

64-bit DMA addresses **9**

A

AMD PCnet32 device **31**

AMD PowerNow! **10**

AMD-V **10**

anti-virus programs
scheduling **26**

B

backups
scheduling **26**

balloon driver
and VMware Tools **25**

ballooning **19**

BIOS

settings **10**

BusLogic virtual storage adapter
using custom driver **30**

C

C1E halt state **10**

checksum offload **9**

CPU

overhead **16**

CPU virtualization
hardware-assisted **7**

E

E1000 device **31**

Enhanced AMD PowerNow! **10**

Enhanced Intel SpeedStep **10**

EPT (extended page tables) **8, 10**

extended page tables **8, 10**

H

HAL

UP vs. SMP **16**

hardware

BIOS settings **10**

hardware version 7 **15**

hardware-assisted MMU virtualization **8**

high-memory DMA **9**

hyper-threading **7, 10**

I

I/O block sizes **30**

Intel E1000 device **31**

Intel SpeedStep **10**

Intel VT-x **10**

J

jumbo frames **9, 32**

K

kernel

UP vs. SMP **16**

M

memory

ballooning **19**

overcommitment **19**

page sharing **19**

reclamation **19**

swapping **19**

memory management unit virtualization **8**

hardware-assisted **7**

MMU virtualization **8**

hardware-assisted **7**

N

nested page tables **8**

NIC morphing **31**

NICs

autonegotiation **9**

duplex **9**

speed **9**

NO_HZ kernels **27, 28, 29**

node interleaving **10**

NPT (nested page tables) **8**

NTP **26**

O

overhead

CPU **16**

P

page sharing **19**

paravirtual timer **27**

paravirtualization **26**

PCnet32 device **31**

Q

queue depth
 virtual SCSI driver **30**

R

rapid virtualization indexing **8, 10**
receive buffers
 insufficient **32**
RVI (rapid virtualization indexing) **8, 10**

S

shadow page tables **8**
SLES
 and paravirtualization **26**
SMP
 virtual **16**
swapping **19**

T

TCP segmentation offload **9**
tickless timer **27, 28, 29**
timekeeping **26**
timing
 within virtual machines **27**
TLB (translation lookaside buffer) **8**
translation lookaside buffer **8**
TSO **9, 31**
Turbo Mode **10**

U

Ubuntu
 and paravirtualization **26**

V

vCPUs
 number of **16**
virtual machine interface **26**
virtual network adapter
 E1000 **31**
 vlance **31**
 vmxnet **31**
virtual SMP **16**
vlance virtual network device **31**
VMI **26**
VMI paravirtual timer **27**
VMware Tools **30**
 and BusLogic SCSI driver **25**
 balloon driver **25**
vmxnet virtual network device **31**
VT-x **10**

W

Windows 7 **16**
Windows Server 2008 **16**
Windows Time Service **26**
Windows Vista **16**