WHITE PAPER

Virtual Machine Monitor Execution Modes in VMware vSphere™ 4.0

vmware®

Table of Contents

Background

Included in VMware® vSphere™ 4.0 is VMware ESX™ 4, the newest generation of a hypervisor deployed in hundreds of thousands of production environments. All versions of ESX comprise two components: the virtual machine monitor (VMM) and the kernel. The monitor is a thin layer that provides virtual x86 hardware to the overlying operating system. The virtual hardware is called a virtual machine and the operating system it runs is called the guest. It is through the monitor that the virtual machine leverages key technologies in the kernel such as memory management, scheduling, and the network and storage stacks.

Prior to the introduction of hardware support for virtualization, the VMM could only use software techniques for virtualizing x86 processors and providing virtual hardware. This software approach, binary translation (BT), was used for instruction set virtualization and shadow page tables for memory management unit virtualization [1]. Today, both Intel and AMD provide hardware support for CPU virtualization with Intel VT-x and AMD-V, respectively. More recently they added support for memory management unit (MMU) virtualization with Intel EPT and AMD RVI. In the rest of this paper, the following are referred as follows: hardware support for CPU virtualization as hardware virtualization (HV), hardware support for MMU virtualization as hwMMU, and software memory management unit virtualization as swMMU.

For some guests and hardware configurations the VMM may choose to virtualize the CPU and MMU using: (a) no hardware support (BT + swMMU), (b) HV and hwMMU (VT-x + EPT), and (c) HV only (VT-x + swMMU). The method of virtualization that the VMware VMM chooses for a particular guest on a certain platform is known as the monitor execution mode or simply monitor mode. On modern x86 CPUs the VMM has an option of choosing from several possible monitor modes. However, not all modes provide similar performance. A lot depends on the available CPU features and the guest OS behavior. VMware ESX identifies the hardware platform and chooses a default monitor mode for a particular guest on that platform. This decision is made by the VMM based on the available CPU features on a platform and the guest behavior on that platform.

VMware recently published a paper that explains different monitor modes, how monitor modes are chosen, and how a monitor mode may significantly impact application performance [2]. Also published were performance white papers comparing the performance of hwMMU versus swMMU on two modern x86 processors: A third generation AMD Opteron 8384 processor [3] and an Intel Xeon E5500 processor [4]. These papers highlight the performance gains observed for a variety of MMU intensive workloads by effectively leveraging hwMMU on these platforms.

In some cases, the default monitor mode may not be optimal for performance. In such cases, the user may choose to override the default monitor mode for that guest. The user can do this by either manually appending a few VMM parameters to the VM configuration file or via the vSphere client GUI. Please refer to the Editing the Virtual Machine Configuration File section for more details.

Initially listed are VMware vSphere 4.0 default monitor modes chosen for many popular guests running modern x86 CPUs. Most workloads perform well under these default settings. In some cases, a user may choose to override the default monitor mode; the next section includes a few examples in which the user may observe performance benefits by doing so. The final section examines two ways of changing the default monitor mode of the virtual machine in vSphere.

Default Monitor Modes

Since 1999, the year VMware released its first product, VMware Workstation, the x86 hardware has undergone many architectural changes, the details of which are beyond the scope of this paper. As virtualization has become ubiquitous, Intel and AMD have increased support for virtual computing. Consequently, the VMware VMM has adopted hardware support for virtualization on x86 CPUs when it is available. Agesen [2] details how the VMware VMM has evolved over time and took advantage of hardware support for virtualization. As mentioned [2], the interaction between guest operating system and available hardware features motivates the monitor mode that the VMware VMM uses by default. Table 1 lists the default monitor modes chosen by vSphere on several Intel processors and Table 2 lists some popular Intel Xeon processors available in the market. Similarly, Table 3 and Table 4 list the default monitor modes for AMD Opteron processors and examples of popular AMD Opteron processors in the market.

Table 1: Default monitor modes on Intel processors

Virtual Machine Configuration	Core i7	45nm Core2 with VT-x	65nm Core2 with VT-x and FlexPriority	65nm Core2 with VT-x and No FlexPriority	P4 with VT-x	EM64T without VT-x	No EM64T
FT enabled	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	Not Runnable	Not Runnable	Not Runnable
64-bit Guests	VT-x + EPT	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	Not Runnable	Not Runnable
VMI enabled(**)	BT + swMMU	BT + swMMU	BT + swMMU	BT + swMMU	BT + swMMU	BT + swMMU	BT + swMMU
OpenServer UnixWare	VT-x + EPT	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	BT + swMMU	BT + swMMU
OS/2	VT-x + EPT	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	Not Runnable	Not Runnable
32-bit Linux 32-bit FreeBSD	VT-x + EPT	VT-x + swMMU	BT + swMMU (*)	BT + swMMU (*)	BT + swMMU (*)	BT + swMMU	BT + swMMU
32-bit Windows: XP, Vista, Server 2003, Server 2008	VT-x + EPT	VT-x + swMMU	VT-x + swMMU	BT + swMMU (*)	BT + swMMU (*)	BT + swMMU	BT + swMMU
Windows 2000, NT, 95, 98, DOS, Netware, 32-bit Solaris	BT + swMMU (*)	BT + swMMU (*)	BT + swMMU (*)	BT + swMMU (*)	BT + swMMU (*)	BT + swMMU	BT + swMMU
Other 32-bit Guests	VT-x + EPT	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	VT-x + swMMU	BT + swMMU	BT + swMMU

(*) When BT is used on an Intel system with VT-x capability, it dynamically switches to VT-x if the guest enters long mode.

(**) In VMI, para-virtualization is used for CPU and MMU virtualization. The guest kernel is statically modified to make hyper-calls into the VMM to execute privileged instructions and perform page table updates [5].

Table 2: Intel processor technology and common server processors

Processor technology (from some columns of Table 1)	Server processors
Core i7	Intel Xeon 5500 series
45nm Core 2 with VT-x	Intel Xeon 5400 series Intel Xeon 7400 series
65nm Core 2 with VT-x and FlexPriority	Intel Xeon 7300 series
65nm Core 2 with VT-x and No FlexPriority	Intel Xeon 5300 series

Table 3: Default monitor modes on AMD processors

Virtual Machine Configuration	Third Generation AMD Opteron and Newer	AMD64 Pre-Third Generation AMD Opteron	No AMD64
FT Enabled	AMD-V + swMMU	Not Runnable	Not Runnable
64-bit Guests	AMD-V + RVI	BT + swMMU	Not Runnable
VMI Enabled(**)	BT + swMMU	BT + swMMU	BT + swMMU
OpenServer UnixWare	AMD-V + RVI	BT + swMMU	BT + swMMU
OS/2	AMD-V +RVI	Not Runnable	Not Runnable
32-bit Linux 32-bit FreeBSD	AMD-V + RVI	BT + swMMU	BT + swMMU
32-bit Windows: XP, Vista, Server 2003, Server 2008	BT + swMMU	BT + swMMU	BT + swMMU
Windows 2000, NT, 95, 98, DOS, Netware, 32-bit Solaris	BT + swMMU	BT + swMMU	BT + swMMU
Other 32-bit Guests	AMD-V + RVI	BT + swMMU	BT + swMMU

(**) In VMI, para-virtualization is used for CPU and MMU virtualization. The guest kernel is statically modified to make hyper-calls into the VMM to execute privileged instructions and perform page table updates [5].

Table 4: Processor technology from AMD Opteron processors

Processor technology (from some columns of Table 3)	Server processors
Six-core AMD Opteron (newer than Third Generation Opteron processors)	AMD Opteron 2400 series AMD Opteron 8400 series
Third Generation Quad-core AMD Opteron	AMD Opteron 2300 series AMD Opteron 8300 series
AMD64 Pre-Third Generation AMD Opteron	AMD Opteron 1200 series AMD Opteron 2200 series AMD Opteron 8200 series

Overriding Default Monitor Mode: When?

For the majority of workloads the default monitor mode chosen by the VMM works best. The default monitor mode for each guest on each CPU has been carefully selected after a performance evaluation of available choices. However, some applications have special characteristics that can result in better performance when using a non-default monitor mode. This section provides a few examples of situations in which a user may wish to override the default monitor mode. Please evaluate the performance of your specific application and use cases against the various execution modes to determine the most suitable option.

Scenario 1: *SPECjbb 2005 on hwMMU capable processors with small pages*

SPECjbb is an industry-standard benchmark for evaluating the performance of server-side Java. SPECjbb runs mostly in a single address space, therefore it has little MMU activity. However due to its broad memory access pattern, it frequently accesses memory addresses not referenced in the translation lookaside buffer (TLB). For most guests the VMM chooses HV + hwMMU as the execution mode on hwMMU capable processors. The hwMMU suffers performance penalties on TLB miss intensive workloads due to the increased page walk latency [3, 4]. VMware strongly recommends using large pages in the guest for such workloads. VMware vSphere 4.0 uses large pages for its own memory to reduce the cost of increased page walk latencies in hwMMU.

If you run SPECjbb or any other highly TLB miss intensive workload with relatively less MMU activity with small pages configured in the guest, then the workload's peformance will suffer with hwMMU as compared to native. This performance loss can be avoided by using swMMU instead of hwMMU. Specifically, this corresponds to using AMD-V and swMMU or BT and swMMU on AMD processors. Similarly Intel VT-x and swMMU or BT and swMMU on Intel processors will improve performance as compared to hwMMU.

Scenario 2: *Running a workload inside 32-bit Windows XP guest (with ACPI Multiprocessor HAL) on Intel Xeon processors with FlexPriority e.g. Xeon 5400s, Xeon 7300s, Xeon 5500s*

For a 32-bit Windows XP guest on Xeon processors, the default execution mode chosen by the VMM for CPU virtualization is Intel VT-x. VMware VMM uses Intel EPT for MMU virtualization when the hardware support is available (e.g., on Intel Xeon 5500 series processors). This guest accesses the APIC's (Advanced Programmable Interrupt Controller) TPR (Task Priority Register) excessively. Prior to the introduction of Intel's FlexPriority feature in Intel VT-x, when the guest would access the TPR, the VMM would interrupt the guest execution, and the guest would access the virtualized TPR instead of the real hardware TPR. On systems without FlexPriority, the VMM uses BT for CPU virtualization in this configuration. With FlexPriority, there is no need for intercepting guest execution as the hardware itself provides a mechanism for storing the virtual TPR state. On systems with FlexPriority the VMM chooses Intel VT-x for CPU virtualization for this configuration. However, even with Intel VT-x and FlexPriority, TPR accesses are slower than native. In BT mode, however, the VMM optimizes the TPR accesses to be better than native performance. Therefore, a few workloads may perform better in BT execution mode than with Intel VT-x on these guests. The only way to be sure that this applies to your situation is to run with both configurations and compare the results.

Scenario 3: *Running SAP in benchmarking mode on AMD third-generation Opteron (and later) or Intel Xeon 5500 Series*

SAP supports a benchmarking configuration to evaluate the peak performance of a machine. In this unsupported configuration, it usually runs in a flat, unprotected memory address space and does not have significant MMU overheads. However, in production deployments, the software runs in a configuration that requires guest kernel services and exhibits significant MMU activity. On the processors mentioned above, the VMM chooses HV and hwMMU on Intel and AMD for most guests. This mode works well in production deployment configurations, minimizing the effect of MMU activity by exploiting hwMMU. One of the benefits of hwMMU is to eliminate hidden page faults and subsequent shadow page table maintenance incurred in swMMU. However, in SAP's benchmarking configuration, page table updates are rare. Thus the swMMU is not a significant source of overhead. In fact, with swMMU, fewer memory accesses are needed per TLB miss, and the benchmark can perform better than on a virtual machine configured with hwMMU where TLB miss costs are higher.

Scenario 4: *Running VMI enabled guests on AMD third-generation Opteron or Intel Xeon 5500 Series*

When a virtual machine is configured to use VMI, the VMM uses para-virtualization for CPU and MMU virtualization [5]. AMD third-generation Opteron processors (and later) and Intel Xeon 5500 processors provide hardware support for MMU virtualization, which may provide better performance than VMI. In this situation, it is recommended to disable VMI. When VMI is disabled, the VMM will use hwMMU, which will give better performance than VMI for most workloads.

The previous performance scenarios are exceptions: specific combinations of workloads, guest operating systems, and hardware that motivate consideration of a non-default monitor mode. As mentioned before, the default VMM execution modes work well and provide good performance for most application, guest, and hardware combinations.

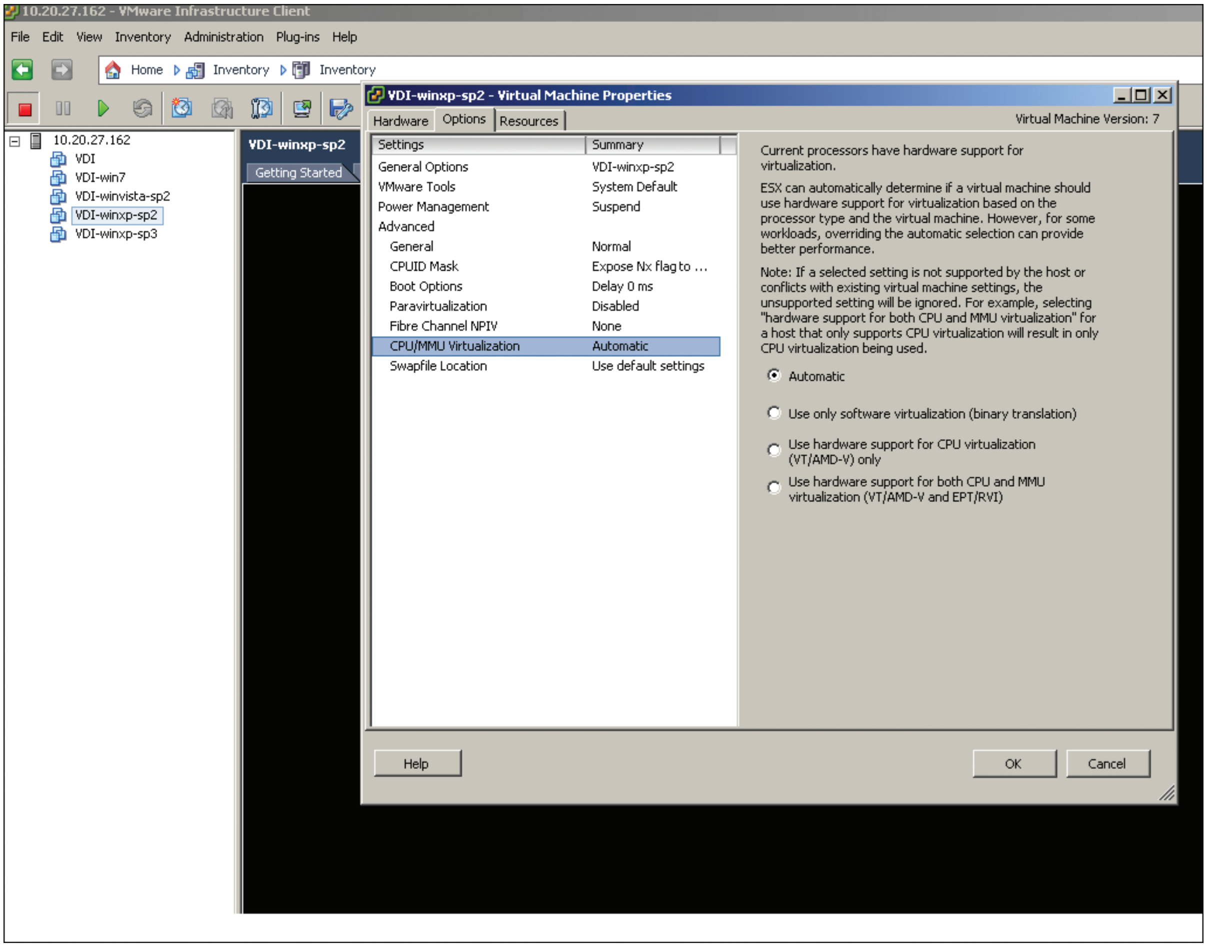
Overriding Default Monitor Mode: How?

There are two ways of overriding the default monitor mode: you can appropriately choose the virtual machine's monitor mode using the vSphere client or manually edit the virtual machine configuration file. If the vSphere client is used to change the monitor mode, the virtual machine configuration files will automatically be modified to incorporate the intended changes.

Using VMware vSphere Client

Open the vSphere client and right-click the virtual machine whose monitor mode you wish to change. Select **Edit Settings** and first click the **Options** tab and then click **CPU/MMU Virtualization** under Advanced options. Figure 1 shows a screenshot with these options.

Figure 1. Selecting monitor modes in the vSphere client



Virtual Machine Version: 7

Settings	Summary
General Options	VDI-winxp-sp2
VMware Tools	System Default
Power Management	Suspend
Advanced	
General	Normal
CPUID Mask	Expose Nx flag to ...
Boot Options	Delay 0 ms
Paravirtualization	Disabled
Fibre Channel NPIV	None
CPU/MMU Virtualization	Automatic
Swapfile Location	Use default settings

Current processors have hardware support for virtualization.

ESX can automatically determine if a virtual machine should use hardware support for virtualization based on the processor type and the virtual machine. However, for some workloads, overriding the automatic selection can provide better performance.

Note: If a selected setting is not supported by the host or conflicts with existing virtual machine settings, the unsupported setting will be ignored. For example, selecting "hardware support for both CPU and MMU virtualization" for a host that only supports CPU virtualization will result in only CPU virtualization being used.

● Automatic

○ Use only software virtualization (binary translation)

○ Use hardware support for CPU virtualization (VT/AMD-V) only

○ Use hardware support for both CPU and MMU virtualization (VT/AMD-V and EPT/RVI)

The client UI presents a four-way choice to enable these two settings for a particular virtual machine. You may choose one of the following options:

- Use automatic (default) for both settings
- Use BT for CPU virtualization and swMMU for MMU virtualization
- Use hardware support for CPU virtualization and swMMU for MMU virtualization
- Use hardware support for CPU and MMU virtualization

The chosen settings are honored by the VMM only if the settings are supported on the current hardware. For example, if **use BT for CPU virtualization** is chosen for a 64-bit guest operating system running on a 64-bit Intel processor, the VMM will choose Intel VT-x for CPU virtualization instead of BT. This is because BT is not supported for 64-bit guests on this processor. Similarly, if **use hardware support for CPU and MMU virtualization** is chosen for any guest on an Intel Xeon 5400 processor, the VMM will choose swMMU for MMU virtualization because the hardware lacks support for Intel's hardware MMU virtualization, Intel EPT. The VMM will choose BT and swMMU when possible if a user specifies **use hardware support for CPU and MMU virtualization** on a system without hardware MMU virtualization support. Please refer to Agesen [2] for more details.

Editing the Virtual Machine Configuration File (*.vmx)

In virtual machine configuration files, the set of modes may be chosen by setting one or both of these settings:

monitor.virtual_exec = software | hardware | automatic

monitor.virtual_mmu = software | hardware | automatic

Choose one of **software**, **hardware**, or **automatic** for each variable. If the setting is set to **software** then the VMM attempts to run the guest without any hardware support. Likewise, if it is set to **hardware** then the VMM forces the use of hardware support. Setting the option to **automatic** chooses the default monitor mode for that option. If a setting is not specified, the effect is the same as automatic. You can choose the following settings:

- If you do not choose any of these settings, then the VMM will use the default monitor mode for your virtual machine. Please see Table 1 and Table 3 for default monitor modes. Making both these settings **automatic** will also behave similarly.
- If you want to use software-only virtualization then set **monitor.virtual_exec = software** and **monitor.virtual_mmu = software**
- If you want to use HV swMMU then set **monitor.virtual_exec = hardware** and **monitor.virtual_mmu = software**
- If you want to choose HV and hwMMU then set **monitor.virtual_exec = hardware** and **monitor.virtual_mmu = hardware**

As stated above, the chosen settings are honored by the VMM only if the setting is supported on the current hardware.

Conclusion

The VMware VMM chooses a default monitor mode to run a specific guest on a particular x86 CPU. This paper makes an effort to familiarize the reader with the default monitor modes chosen by VMware vSphere 4.0 for common guests on modern x86 server processors. Most workloads perform well under the default monitor mode chosen by VMware vSphere 4.0. However, in some cases the workload characteristics may drive it to perform better in a non-default monitor mode. This paper provides the reader with a few examples of such situations and guides them on how they can change the default monitor mode.

References

1. **Keith Adams and Ole Agesen**. *A comparison of software and hardware techniques for x86 virtualization.* San Jose, USA: ACM, 2006. Architectural Support for Programming Languages and Operating Systems.
2. **Ole Agesen.** *Software and Hardware Techniques for x86 Virtualization.* Palo Alto CA,VMware Inc. , 2009.
3. **Nikhil Bhatia.** *Performance Evaluation of AMD RVI.* Palo Alto CA, VMware, 2008.
4. **Nikhil Bhatia.** *Performance Evaluation of Intel EPT.* Palo Alto CA, VMware, 2009.
5. **VMware Inc.** *Performance of VMware VMI.* Palo Alto CA, VMware, 2008.

About the Author

Nikhil Bhatia is a Performance Engineer at VMware. In this role, his primary focus is to evaluate and help improve the performance of the VMware Virtual Machine Monitor. Prior to VMware, Nikhil was a researcher at Oak Ridge National Laboratory (ORNL) in the Computer Science and Mathematics Division. Nikhil received a Master of Science in Computer Science from the University of Tennessee, where he specialized in tools for performance analysis of High Performance Computing (HPC) applications.

Acknowledgements

The author would like to thank the VMM developers for their tireless efforts in developing and optimizing the VMM. The author would also like to thank members of the VMware VMM and Performance groups who have helped shape the document to its current state: Ole Agesen, Ken Barr, Jeff Buell, Jim Mattson, Kiran Tati, and Aravind Pavaluri.

Virtual Machine Monitor Execution Modes in VMware vSphere 4.0
Source: Technical Marketing, SD
Revision: 20090812

VMware, Inc. 3401 Hillview Ave Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com

vmware®