VMworld2005
virtualizenow

VMware and CPU Virtualization Technology

Jack Lo
Sr. Director, R&D

VMworld2005 virtualizenow

This presentation may contain VMware confidential information.

Copyright © 2005 VMware, Inc. All rights reserved. All other marks and names mentioned herein may be trademarks of their respective companies.

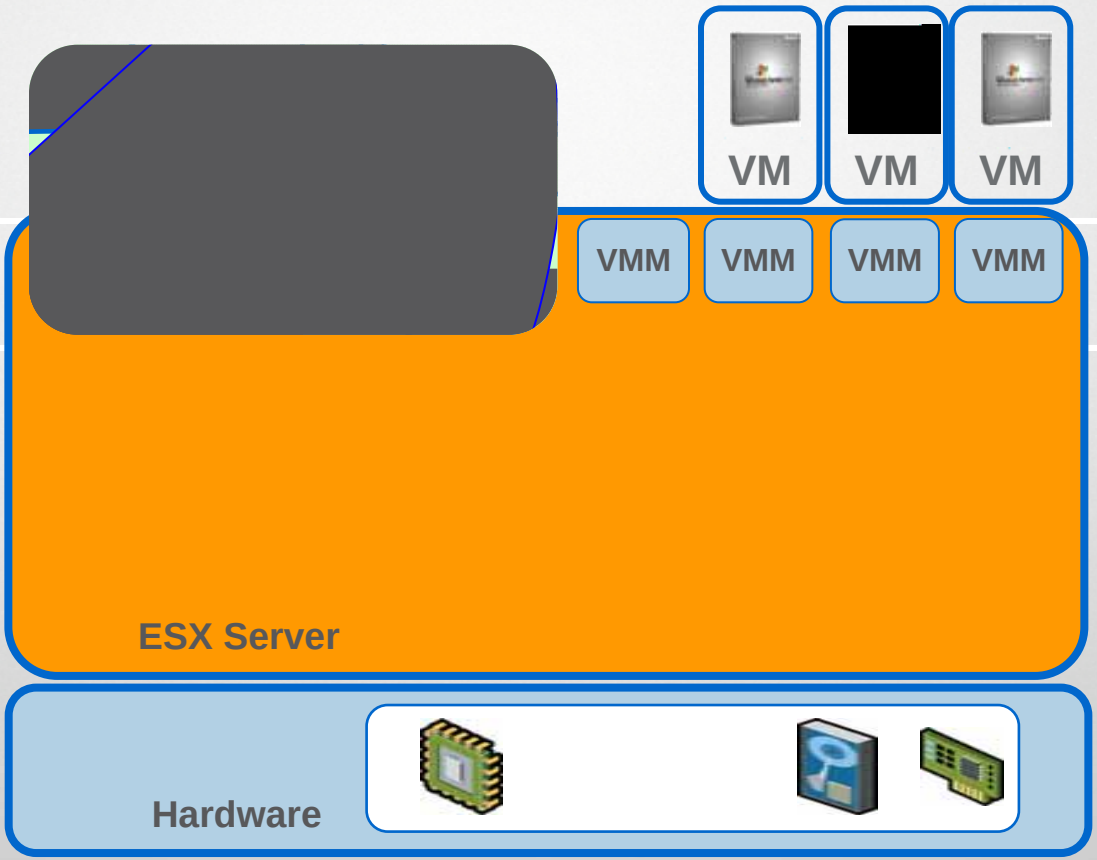
Overview

- Emerging technologies that impact CPU virtualization
  - Hardware assist (VT-x/Pacifica)
  - 64-bit computing
  - OS assist (paravirtualization)
- Today's talk:
  - Share our perspective on emerging technologies

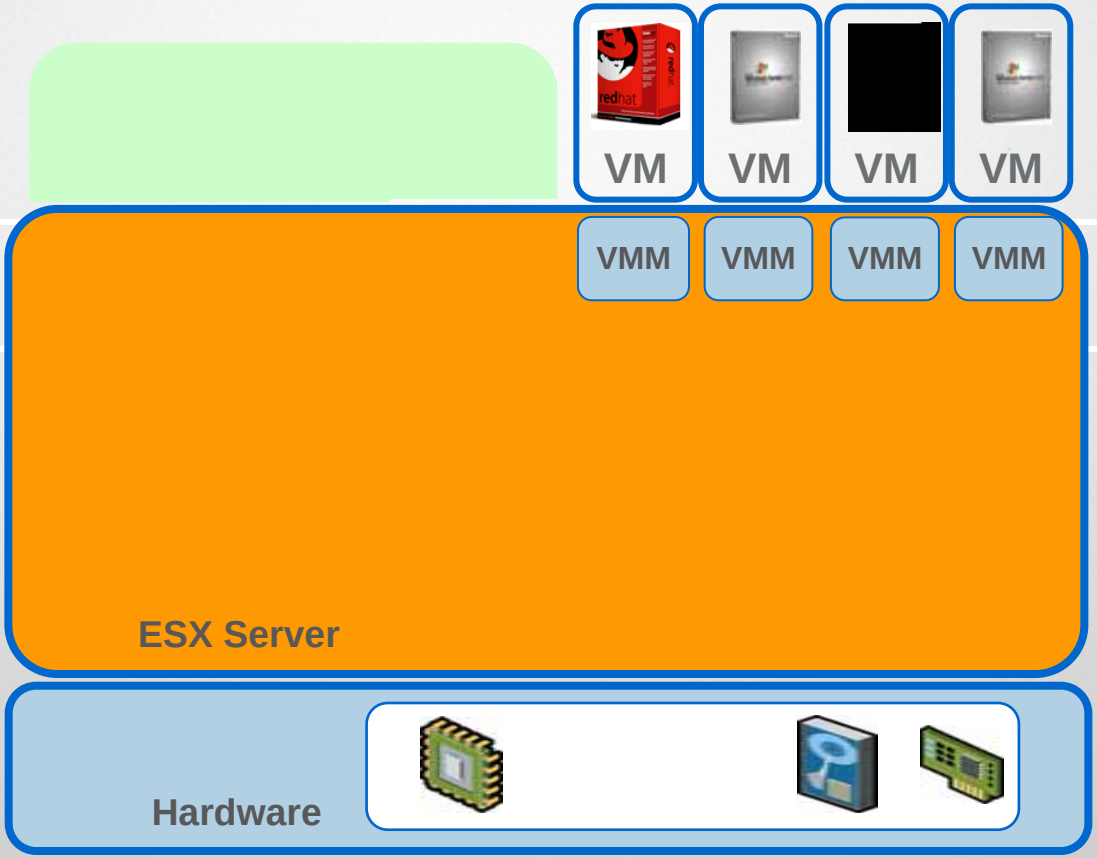
Agenda

- CPU virtualization technology overview
  - Virtualizing the x86 architecture
- Trend No. 1: Hardware assist
  - VT-x and Pacifica
- Trend No. 2: 64-bit computing
  - Benefits of 64-bit architecture
  - 64-bit guest support
- Trend No. 3: OS assist
  - VMware and paravirtualization

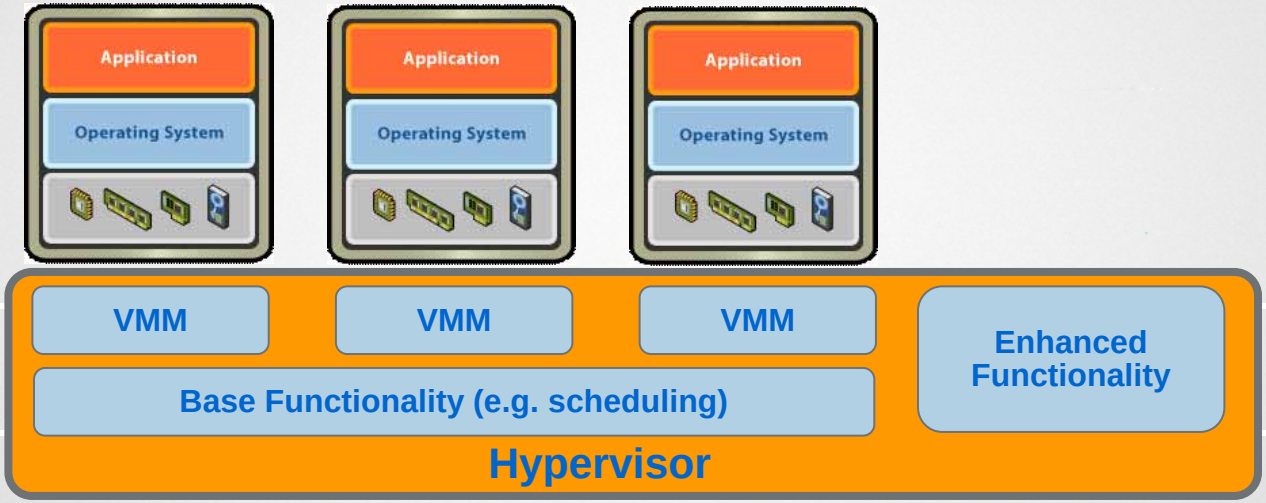
Full Virtualization Software Stack



Today's Focus



Virtualization SW Terminology



- Virtual Machine Monitor (VMM)
  - SW component that implements virtual machine hardware abstraction
  - Responsible for running the guest OS
- Hypervisor
  - Software responsible for hosting and managing virtual machines
  - Run directly on the hardware
  - Functionality varies greatly with architecture and implementation

CPU Virtualization

- Three components to classical virtualization techniques
- Many virtualization technologies focus on handling privileged instructions

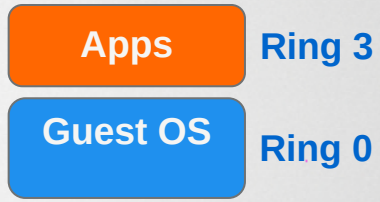
Privileged instruction virtualization	De-privileging or ring compression to handle privileged instructions

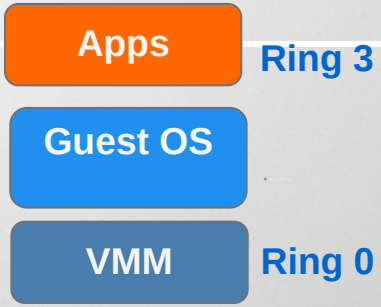
Memory virtualization	Memory partitioning and allocation of physical memory

Device and I/O virtualization	Routing I/O requests between virtual devices and physical hardware

Handling Privileged Instructions

- In traditional systems
  - OS runs in privileged mode
  - OS "owns" the hardware
  - Application code has less privilege
- VMM needs highest privilege level for isolation and performance
- Traditional VMM relies on "ring compression" or "de-privileging"
  - Run privileged guest OS code at user-level
  - Privileged instructions trap, and emulated by VMM



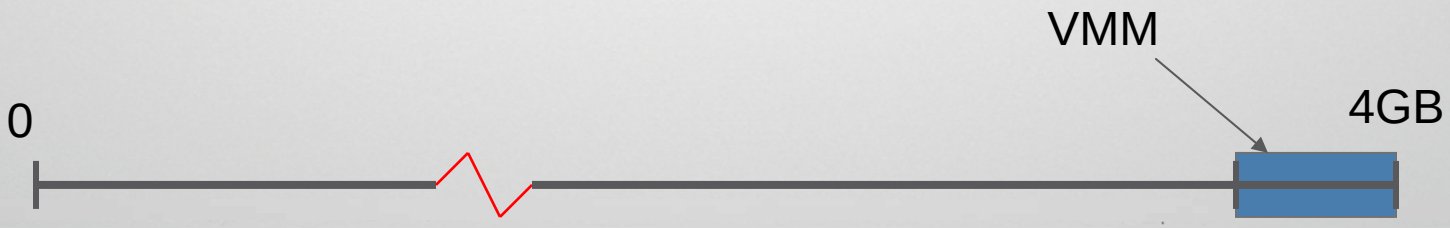


Virtualizing x86 Architecture

- De-privileging not possible with x86!
  - Some privileged instructions have different semantics at user-level: “non-virtualizable instructions”
- VMware uses direct execution and binary translation (BT)
  - BT for handling privileged code
  - Direct execution of user-level code for performance
  - Any unmodified x86 OS can run in virtual machine

Protecting the VMM

- Need to protect VMM and ensure isolation
  - Protect virtual machines from each other
  - Protect VMM from virtual machines
- VMware relies on segmentation hardware to protect the VMM
  - VMM lives at top of guest address space
  - Segment limit checks catch writes to VMM area



Agenda

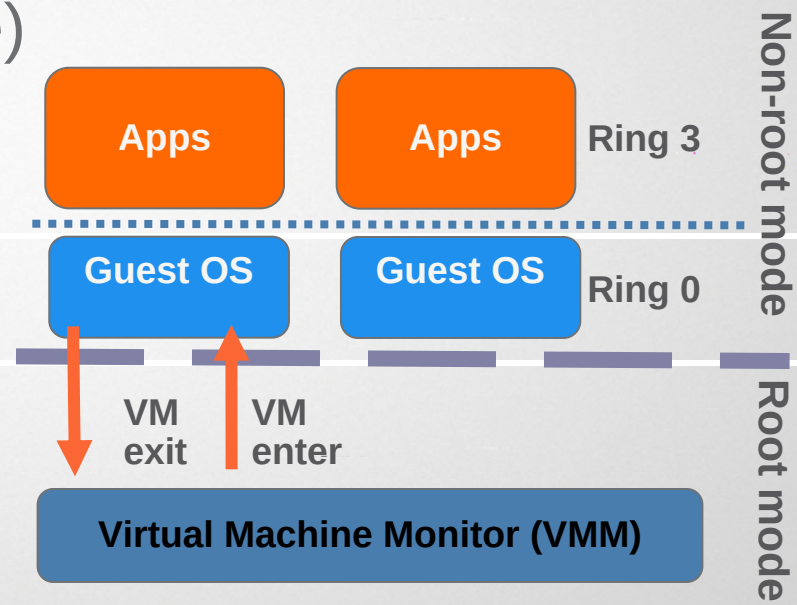
- CPU virtualization technology overview
  - Virtualizing the x86 architecture
- **Trend No. 1: Hardware assist**
- Trend No. 2: 64-bit computing
- Trend No. 3: OS assist

Trend No. 1: Hardware Assist

- CPU vendors are embracing virtualization
  - Intel Virtualization Technology (VT-x)
  - AMD Pacifica
- These CPU technologies are a series of enhancements to aid virtualization SW
  - Initially focused on handling non-virtualizable instructions
  - Use a trap-and-emulate model
  - Alternative to using binary translation
- But hardware assist does not eliminate need for VMware technology

VT-x/Pacifica Overview

- Key feature is new CPU execution mode (root mode)
  - VMM executes in root mode
  - Allows x86 virtualization without binary translation or paravirtualization
  - Guest state stored in Virtual Machine Control Structures (VT-x) or Virtual Machine Control Block (Pacifica)



Limitations of Hardware Assist

- Initial VT-x/Pacifica hardware does not include all components of CPU virtualization solution
- VT-x requires small emulator for real mode code
- Memory virtualization support lacking
  - Not in VT-x; implementation-dependent for Pacifica
  - Memory virtualization is key to performance!
- No device virtualization support

	Hardware Assist
Privileged instructions	Yes
Memory virtualization	No
Device and I/O virtualization	No

Future of Hardware Assist

- CPU vendors will add more hardware capabilities in future
  - Memory virtualization (Nested paging, EPT)
- VMware software will evolve to incorporate support for these new technologies
  - Adopt technologies as they enable new capabilities

	Hardware Solution
Privileged instructions	VT-x, Pacifica
Memory virtualization	Extended Page Tables/Nested Paging
Devices and I/O	Intelligent Devices

Trend No. 2: 64-bit Computing

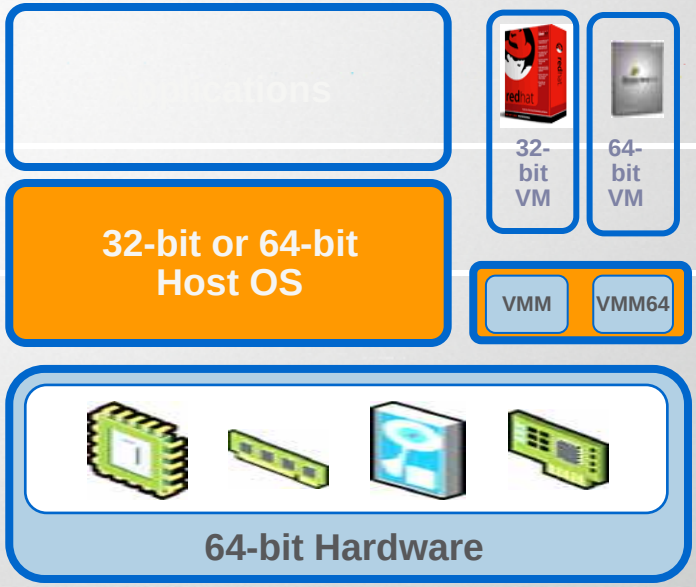
- Progression of the x86 architecture
  - 16-bit: 8086/8088 (1978)
  - 32-bit: 80386 (1985)
  - 64-bit: x86-64 (2003): a.k.a. AMD64, x64
- x86-64 architecture brings 64-bit computing to industry-standard systems
  - Provides compatibility mode to run 32-bit x86 applications
  - Extensions to x86 architecture

64-bit Transition Has Already Begun

- Apps exhausting limits of 32-bit address space
  - Consuming 1 bit of address space / year
  - Databases, Java app servers, other threaded applications
- Most new CPUs are 64-bit enabled
  - AMD64, EM64T
- Major OSes have been ported
  - Windows, Linux, Solaris 10, etc.
- Applications are being ported
  - Databases, app servers, development tools, games, etc.

Virtualization And x86-64

- Potential questions about 64-bit transition
  - Do my apps run in 64-bit OS?
  - Have drivers been ported?
  - Are the 64-bit OSes robust?
- The solution: virtualization!
  - Great aid for 64-bit transition
  - Easy way to evaluate new OSes
  - Can run 64-bit guest OSes on 32-bit host OS on 64-bit hardware)!



Challenges of Virtualizing x86-64

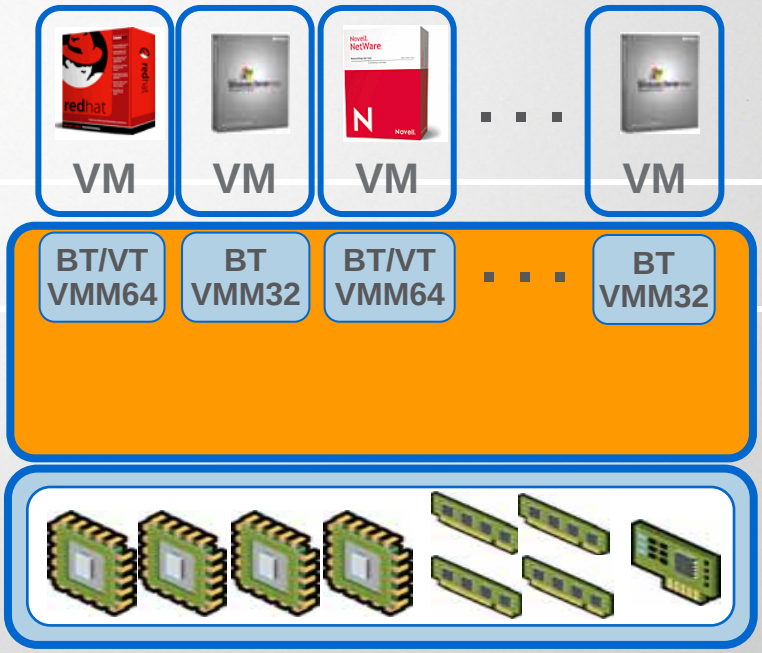
- Initial AMD64 architecture did not include segmentation in 64-bit mode
  - Segmentation also missing from EM64T

How do we protect the VMM?

- 64-bit guest support requires additional hardware assistance
  - Segment limit checks available in 64-bit mode on newer AMD processors
  - VT-x can be used to protect the VMM on EM64T
    - Requires trap-and-emulate approach instead of BT

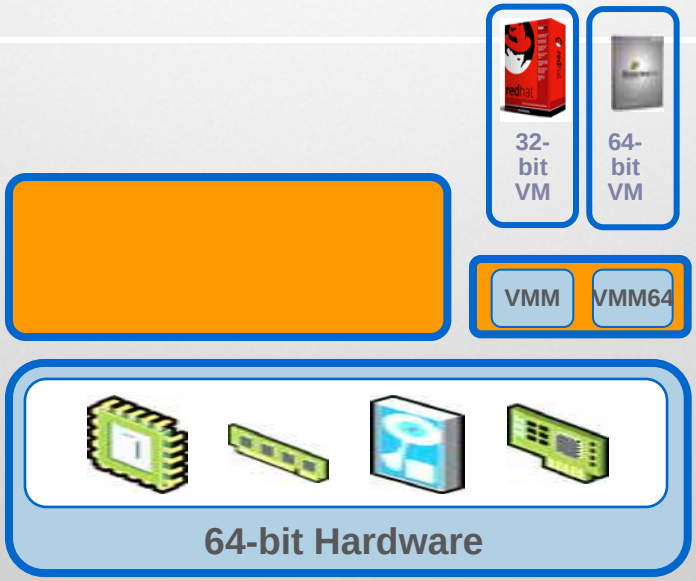
Flexible VMM Architecture

- Flexible "multi-mode" VMM architecture
  - Separate VMM per virtual machine
  - 32-bit: BT VMM
  - 64-bit: BT or VT/Pacifica VMM depending on hardware
- Select mode that achieves best workload-specific performance
- Same VMM architecture for ESX Server, GSX Server, Workstation and ACE



64-bit Guests And WS 5.5

- Workstation 5.5 enables 64-bit guests
  - Currently in beta
- Simultaneously run 32-bit and 64-bit guests
- Runs on 32-bit and 64-bit host OSes



Requirements For 64-bit Guests

- Newer hardware required for 64-bit guest support
  - AMD Opteron Rev. E or later
  - AMD Athlon64 Rev. D or later
  - Intel VT-enabled processor
- How to determine that you have a 64-bit capable system?
  - Workstation 5.5 will automatically check to see if your CPU meets the requirements
  - CPU check utility also available for download on WS5.5 beta web page
  - **http://www.vmware.com/products/beta/ws/**

Trend No. 3: OS Assist

- Three alternatives for handling non-virtualizable instructions
  - Binary translation
  - Hardware assist
  - OS assist or paravirtualization

Performance	**Good**	**Average**
VMM sophistication	**High**	**Average**

Paravirtualization

- Paravirtualization can address same problem as hardware assist
  - Modify the guest OS to remove non-virtualizable instructions
  - Export a simpler architecture to OS
  - Cannot support unmodified OSes (e.g. Windows 2000/XP)
  - Paravirtualization not limited to CPU virtualization
  - Higher performance possible
  - Relatively easy to add paravirtualization support: very difficult to add binary translation

	Binary Translation	Hardware Assist	Para-virtualization
Compatibility	**Excellent**	**Excellent**	**Poor**
Performance	**Good**	**Average**	**Excellent**
VMM sophistication	**High**	**Average**	**Average**

Paravirtualization Challenges

- XenoLinux paravirtualization approach unsuitable for enterprise use
  - Relies on separate kernel for native and in virtual machine
  - Guest OS and hypervisor tightly coupled
  - Tight coupling inhibits compatibility
  - Changes to the guest OS are invasive
  - Inhibits maintainability and supportability
  - Guest kernel must be recompiled when hypervisor is updated
- How can we deliver paravirtualization for enterprise customers?

VMI Paravirtualization API

- VMware proposal: Virtual machine Interface API
  - VMI provides maintainability & stability
  - API supports low-level and higher-level interfaces
  - Allows same kernel to run natively and in a paravirtualized virtual machine: “transparent paravirtualization”
  - Allows for replacement of hypervisors without a guest recompile
  - Preserve key virtualization functionality: page sharing, VMotion, etc.
- We are gathering feedback on the API from many kernel developers and OSVs
  - http://www.vmware.com/vmi
  - http://www.vmware.com/standards/hypercalls.html

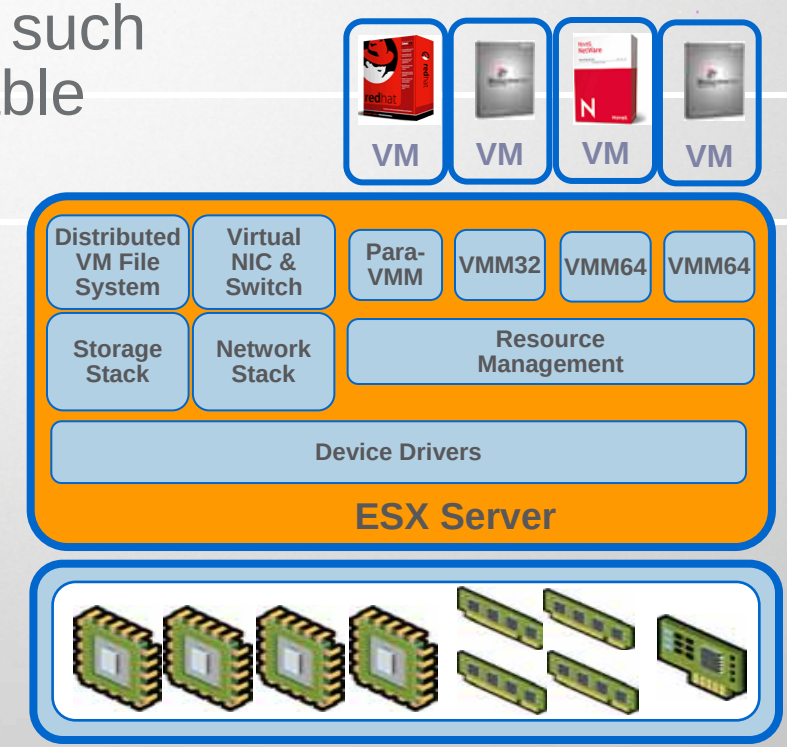
VMI Paravirtualization

- VMI approach to paravirtualization improves compatibility
- API need not compromise performance compared to invasive paravirtualization

	Binary Translation	Hardware Assist	Para-virtualization
Compatibility	**Excellent**	**Excellent**	**Good**
Performance	**Good**	**Average**	**Excellent**
VMM sophistication	**High**	**Average**	**Average**

VMware And Paravirtualization

- VMware will support paravirtualized Linux OSes
  - Another guest type when such OS's commercially available
- Flexible architecture
  - Use most efficient technique for the guest OS type
  - BT, VT/Pacifica, or paravirtualization



Summary

- 64-bit transition happening now
  - Virtualization can assist with transition
  - 64-bit guests supported in WS5.5
- VMware provides flexible architecture to support emerging virtualization technologies
  - Multi-mode VMM utilizes binary translation, hardware assist and paravirtualization
  - Select best operating mode for the workload
- VMware will support paravirtualized guests as they appear in enterprise distributions
  - VMI offers superior maintainability/flexibility
  - Performs as well as invasive paravirtualization

VMworld2005
virtualizenow

PAC346
VMware and CPU Virtualization Technology

Jack Lo
Sr. Director, R&D

Backup slides

Performance of Binary Translation

- BT provides many performance optimization opportunities
  - Fault elimination
    - Avoid costs of repeated virtual machine exits
    - Binary translator identifies faulting instructions and replaces them with special translations
  - Jump directly to appropriate handlers without an expensive fault
    - Guest and VMM share an address space: reduces context switch costs