The Future of Virtualization Technology

Stephen Alan Herrod
VP of Technology
VMware
Agenda

• Virtualization Today
• Technology Trends and the Future Datacenter
• Future directions
  • CPU Virtualization
  • I/O Virtualization
  • Virtual appliances
• Conclusions
X86 Server Virtualization Basics

Before Server Virtualization:

- Single OS image per machine
- Software and hardware tightly coupled
- Running multiple applications on same machine often creates conflict
- Underutilized resources
Before Server Virtualization:
- Single OS image per machine
- Software and hardware tightly coupled
- Running multiple applications on same machine often creates conflict
- Underutilized resources

After Server Virtualization:
- Virtual machines (VMs) break 1-to-1 dependency between OS and HW
- Manage OS and application as single unit by encapsulating them into VMs
- Strong isolation between VMs
- Hardware-independent: they can be provisioned anywhere
X86 Server Virtualization Architectures

• Hosted Architectures
  • Install as applications on Windows/Linux with small context switching driver
  • Leverage host IO stack and resource management
  • Examples include VMware Workstation, VMware Server, Microsoft Virtual PC, Microsoft Virtual Server, ...

• Bare-metal Architectures
  • “Hypervisor” installs directly on hardware
  • Approach acknowledged as direction for datacenter
  • VMware ESX Server, Xen, Microsoft Viridian
Bare-metal Example: VMware ESX Server

Service Console

Resource Management

High-performance storage and network virtualization

Device Drivers

VMkernel Hardware Interface

VMkernel Hypervisor

Services

SDK and Management Agents

VMX VMX VMX VMX

VM VM VM VM

VMM VMM VMM VMM

Hardware
Benefits Grow with Distributed Virtualization

- Distributed file system allows multiple machines to see VMs
- Treat servers as a unified pool of resources
- Live migration (VMotion) of VMs between servers
  - Encapsulation and HW independence is key to this!
Managing Distributed Virtualization

- Centralized management of hardware and VMs is key
  - Inventory of hardware and virtual machines (and their mappings)
  - Historic performance information
  - Remote consoles and devices
  - Drive VMotion between servers via drag-and-drop
Distributed Virtualization Benefit: VMware DRS

With management, performance information, and VMotion:

- Input service level “rules” for each virtual machine
- Virtual Center uses VMotion to continuously optimize based on workload
- Reacts to adding or removing hosts from the cluster
Distributed Virtualization Benefit: VMware HA

With management, heartbeat, shared storage

- Losing a physical server means fewer resources, not lost virtual machines
  - Impacted virtual machines are restarted on remaining hosts
  - Placement optimized by global scheduler
Common Virtualization Uses Today

Server Consolidation and Containment – Eliminate server sprawl by deploying systems into virtual machines

Infrastructure Provisioning – Reduce the time for provisioning new infrastructure to minutes with sophisticated automation capabilities. Like copying a file!

Business Continuity – Reduce the cost and complexity of business continuity by encapsulating entire systems files that can be replicated and restored onto any target server

Test and Development – Rapidly provision and re-provision test and development servers; store libraries of pre-configured test machines

Enterprise Desktop – Secure unmanaged PCs. Alternatively, provide standardized enterprise desktop environments hosted on servers.

Legacy Application Re-hosting – Migrate legacy operating systems and software applications to virtual machines running on new hardware for better reliability
Agenda

• Virtualization Today

• Technology Trends and the Future Datacenter

• Future directions
  • CPU Virtualization
  • I/O Virtualization
  • Virtual appliances

• Conclusions
Technology Trends in the Datacenter

• Multi-core CPUs (ISCA Session #4)
  • 16+ CPUs/cores per server
  • Increasing NUMA-ness

• 64-bit addressing
  • Enables huge amounts of physical memory

• Cooling and power costs soaring (ISCA Session #3)
  • Power-aware CPUs, servers, and racks

• Converged I/O fabrics
  • Shared high-speed interface to network and storage

• Network-based, virtualized storage
  • Stateless servers with flexible I/O connections
Virtualization is Key to Exploiting Trends

- Allows most efficient use of the compute resources
  - Few apps take advantage of 16+ CPUs and huge memory as well as virtualization
  - Virtualization layer worries about NUMA, not apps
- Maximize performance per watt across all servers
  - Run VMs on minimal # of servers, shutting off the others
  - Automated, live migration critical:
    - Provide performance guarantees for dynamic workloads
    - Balance load to minimize number of active servers
- Stateless, Run-anywhere Capabilities
  - Shared network and storage allows flexible mappings
  - Enables additional availability guarantees
Vision: The Fully Virtual Datacenter

Pool of Hardware
Vision: The Fully Virtual Datacenter

Dynamically-Mapped Services

Pool of Hardware

16 GHz
16 GB
Tier-2 Storage

Test Clients

Test/Dev Pool

App
Win 2K
RH 7.3

Apache
RH EL3

App
Win2k

Apache
RH 7.3

DB
NT4

Win2k

16 GHz
16 GB

Storage
Vision: The Fully Virtual Datacenter

**Dynamically-Mapped Services**

Internet

Intranet

Private Networks

Production Pool

Test/Dev Pool

Pool of Hardware

**32 GHz 64 GB Tier-1 Storage**

**16 GHz 16 GB Tier-2 Storage**

**Vision:** The Fully Virtual Datacenter
Agenda

- Virtualization Today
- Technology Trends and the Future Datacenter
- Future directions
  - CPU Virtualization
  - I/O Virtualization
  - Virtual appliances
- Conclusions
Hardware and OS Support on the Way

• New HW technologies provide explicit support
  • Increased CPU assist coming via Intel VT* and AMD SVM
    • Uber context switches
    • Better trapping
    • Hidden memory for hypervisor/vmm
  • I/O MMU’s and virtualization-accelerating I/O devices

• Operating systems become virtualization-aware
  • Provide hints to the hypervisor about what is going on.
  • Skip expensive-to-virtualize operations altogether
  • Friendly licensing terms!

How do we best leverage this support?
CPU Virtualization

- Service Console
  - I/O Stack
  - Device Drivers

- VMkernel
  - High-performance storage and network virtualization

- Hardware

- VM with Device Drivers

- VMM with Device Drivers
Virtual Machine Monitor (VMM)

- One for each VM
- Connects virtual resources to physical resources
- Must guarantee
  - Isolation of the VM
  - Compatibility
  - High performance
Approach to CPU Virtualization

- Basic idea: directly execute code until not safe
- Handling unsafe code
  - Trap & emulate: classic approach, easier w/CPU support
  - Avoid unsafe code, call into VMM: paravirtualization
  - Dynamically transform to safe code: binary translation

<table>
<thead>
<tr>
<th>Tradeoffs among the methods</th>
<th>Trap &amp; Emulate</th>
<th>Para-virtualize</th>
<th>Binary Translate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>Average</td>
<td>Excellent</td>
<td>Good</td>
</tr>
<tr>
<td>Compatibility</td>
<td>Excellent</td>
<td>Poor</td>
<td>Excellent</td>
</tr>
<tr>
<td>Implementation</td>
<td>Average</td>
<td>Average</td>
<td>Hard</td>
</tr>
<tr>
<td>Extra Capabilities</td>
<td>Standard</td>
<td>Many</td>
<td>Many</td>
</tr>
</tbody>
</table>
Our Approach to CPU Virtualization

• We expect multiple, simultaneously-running VMMs
  • 32-bit BT VMM
  • 64-bit VT/SVM VMM
  • Paravirtualized VMM

• Use most efficient method for the hardware and guest type!

The best choices will change as HW support matures and paravirtualization APIs become standard.
Agenda

• Virtualization Today
• Technology Trends and the Future Datacenter
• Future directions
  • CPU Virtualization
  • I/O Virtualization
  • Virtual appliances
• Conclusions
I/O Virtualization

- Hosted/Split I/O
  - VMX
  - VMX
  - VMX
  - VMX

- Device Drivers

- Resource Management

- High-performance storage and network virtualization

- Device Drivers

- Direct I/O

- Service Console

- VMkernel

- Hardware
I/O Virtualization

- Full virtualization of I/O devices. Guests see standard devices regardless of physical machine
  - LSI SCSI Controller
  - E1000 NIC
- Virtual devices mapped to actual devices via
  - hosted/split I/O, or
  - direct I/O
- A third option in the future: passthrough
Hosted/Split I/O Virtualization

• IPC between frontend and backend
  • involves context switch and scheduling delays unless run on dedicated CPU
• Utilize drivers for Driver VM’s OS type
Direct I/O Virtualization

- System call between frontend and backend
  - Backend can run on any CPU
- Utilize VMkernel drivers
  - Linux compatibility layer (locks, memory allocation)
Hosted/Split vs. Direct I/O

- **Performance**
  - Hosted/split uses IPC and generally requires dedicated CPUs
  - Direct I/O generally more efficient and scalable

- **Compatibility**
  - Hosted/split provides easier reuse of device drivers
  - Both require full qualification of drivers for unique workload patterns

- **Fault Isolation**
  - Hosted/split provides additional isolation with drivers in a VM
  - Direct I/O can take advantage of sandboxing and other techniques
  - Both require I/O MMU to provide true isolation
I/O Models and Scalable performance

More Cores = More VMs

IO Backends in shared Service Console become bottleneck
I/O Models and Scalable performance

Choice: Multiple service consoles or break out VMX

Service Console/Driver VM

High-performance storage and network virtualization

Device Drivers

Resource Management

POSIX API

VMX VMX VMX VMX

VM VM VM VM

VMX VMX VMX...

API

Services

Choice: Multiple service consoles or break out VMX
Scaling on 8-way Systems

8-CPU DL-760
16GB RAM

VMX processes in shared Service Console

VMX as processes On VMkernel

Windows 2000 Boot Time (s)

Number of idle Windows 2000 guests
Passthrough I/O Virtualization

- Safely export hardware all the way to the guest
- Let guest OS driver directly drive the device
- VMkernel needed for set up and interrupt routing

Some Use Cases
- Performance (TOE)
- Special HW (USB, TPM, 3D)
- HW-level IO Isolation

Challenges
- Isolation
- HW Independence
Hardware Support for Passthrough

• To preserve capabilities, extra support required:
  • Isolation: I/O MMU to protect VMs from rogue DMA!
  • VMotion and Machine Independence:
    • Extracting and restoring device state (VMotion to same hardware)
    • Standardize device abstraction to VM (VMotion anywhere)
  • Memory over-commitment:
    • Device supports demand paging of memory it accesses
  • Device sharing:
    • Export multiple logical interfaces (e.g., WWNs via NPIV)
    • Track different I/O streams (e.g., tagged network queues)
Summary of I/O Virtualization Tradeoffs

• Use hosted/split I/O for compatibility
  • Peak performance will require dedicated CPUs (Power)
  • Multiple Driver VMs for scalability

• Use direct I/O for top performance
  • Requires some driver porting
  • Improved fault isolation through use of I/O MMUs, sandboxing

• Use passthrough I/O for performance, fault isolation, specialized HW, and compatibility
  • Requires HW support to preserve virtualization functionality
Agenda

• Virtualization Today
• Technology Trends and the Future Datacenter
• Future directions
  • CPU Virtualization
  • I/O Virtualization
  • Virtual appliances
• Conclusions
Virtualization Changes Traditional Boundaries

- OS Coupled With App
- Virtualization Coupled With Hardware
If virtualization is everywhere, could these new boundaries inspire a new distribution model for software?
Virtual Appliances

• All the benefits of a traditional computing appliance without the cost and complexity
• Pre-configured, purpose-built virtual device
• Pre-installed and pre-configured OS & application
• Limited configuration/customization exposed to user
• Simple installation and setup
• Doesn’t require dedicated machine

The Virtual Appliance approach provides a turn-key solution to complex software distribution
Before / After Virtual Appliances

**Before**
- One-to-one ratio of function to device
- Support from multiple sources
- Inefficient utilization of hardware

**After**
- Consolidate to save space/power
- Hardware support from preferred vendor
- Efficient utilization of hardware
## Benefits Matrix

<table>
<thead>
<tr>
<th>Feature</th>
<th>Traditional Software</th>
<th>Hardware Appliance</th>
<th>Virtual Appliance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build and test on a controlled platform</td>
<td>✗ No</td>
<td>✓ Yes</td>
<td>✓ Yes</td>
</tr>
<tr>
<td>Simple plug-and-play installations</td>
<td>✗ No</td>
<td>✓ Yes</td>
<td>✓ Yes</td>
</tr>
<tr>
<td>Ability to tightly control access to underlying OS</td>
<td>✗ No</td>
<td>✓ Yes</td>
<td>✓ Yes</td>
</tr>
<tr>
<td>Inexpensive to distribute to customers</td>
<td>✓ Yes</td>
<td>✗ No</td>
<td>✓ Yes</td>
</tr>
<tr>
<td>Works with existing x86 hardware</td>
<td>✓ Yes</td>
<td>✗ No</td>
<td>✓ Yes</td>
</tr>
<tr>
<td>Low support cost and simple support logistics</td>
<td>✓ Yes</td>
<td>✗ No</td>
<td>✓ Yes</td>
</tr>
<tr>
<td><strong>Easy, quick out-of-box experience for pilots, POC &amp; Demonstrations</strong></td>
<td>✗ No</td>
<td>✗ No</td>
<td>✓ Yes</td>
</tr>
<tr>
<td>Availability (easy backup and restore, easy recovery on new hardware, easy disaster recovery)</td>
<td>✗ No</td>
<td>✗ No</td>
<td>✓ Yes</td>
</tr>
<tr>
<td>Ability to scale up and down as needed without downtime</td>
<td>✗ No</td>
<td>✗ No</td>
<td>✓ Yes</td>
</tr>
<tr>
<td>Provides clustering capabilities without special code</td>
<td>✗ No</td>
<td>✗ No</td>
<td>✓ Yes</td>
</tr>
</tbody>
</table>
VM Library: Virtual Appliance Examples

- More than 200 submissions to recent contest!
- Browser Virtual Appliance
  - Very lean Ubuntu Linux installation + VM console access to Firefox
  - Internet browsing inside ‘contained’ environment of a Virtual Appliance
- Oracle RAC 10g on RHEL
  - Pre-installed and pre-configured to save time
  - Can see cluster behavior on a single machine
- Kid-safe computing
  - Web filter, replacement shell, squid URL blocking
- Voice Over IP (Asterisk@Home, sipX)

Free virtualization layer is a key enabler ala Adobe Reader
## Virtual Appliances

### Filter by Appliance Type:
- **All Appliances**
- Partner Virtual Appliance
- Community Virtual Appliance
- Challenge Entry

### Filter by Solution Area:
- **All Areas**
- Administration
- Application/Web Server
- Communications

### Filter by Target User:
- **All Users**
- Desktop User
- Advanced Desktop User
- Developer/Tester

For multiple selection, CTRL + Click

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
<th>Size</th>
<th>Rating</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personal Backup Appliance</td>
<td>Backup, Restore, or Clone the Hard Disk Drive of any Computer or Virtual Machine over the Network!</td>
<td>384 MB</td>
<td>★★★★★</td>
<td>06/19/2006</td>
</tr>
<tr>
<td>Oracle Desktop Data Center</td>
<td>Oracle Database 10g with Oracle Real Application Clusters</td>
<td></td>
<td>★★★★★</td>
<td>06/19/2006</td>
</tr>
<tr>
<td>JanusVM</td>
<td>Internet Privacy Appliance</td>
<td>~ 38 MB</td>
<td>★★★★★</td>
<td>06/19/2006</td>
</tr>
<tr>
<td>Internet Content Filter</td>
<td>The Internet Content Filter uses proxy server technology to prevent families from accessing inappropriate web sites.</td>
<td>110 MB</td>
<td>★★★★★</td>
<td>06/19/2006</td>
</tr>
<tr>
<td>OSCAR Cluster Headnode</td>
<td>Cluster headnode ready to deploy a Fedora Core 5 based image. Includes tools for managing, administering and running a cluster.</td>
<td>1464 MB</td>
<td>★★★★★</td>
<td>06/19/2006</td>
</tr>
<tr>
<td>Rails Appliance</td>
<td>Rails development environment with lighttpd, fcgi, mysql, svn, ruby, rails, and capistrano</td>
<td>387 MB</td>
<td>★★★★</td>
<td>06/19/2006</td>
</tr>
</tbody>
</table>
Summary

• Virtualization is key to the future datacenter
  • Hardware trends require it for efficiency
• Examined two key areas
  • CPU virtualization
  • I/O virtualization
• Introduced the concept of virtual appliances
  • A new means of software distribution
Calls to Action

• CPU Vendors
  • Continue the progress! (nested page table support, reduced latencies)

• I/O Device Vendors
  • Build required hardware support for pass-through
  • Wide open space for research and start-ups

• Software Developers
  • Support open, standardized paravirtualization interfaces to avoid fragmentation and encourage hypervisor competition
  • Consider the benefits of virtual appliances

Everyone: Help deliver the benefits of virtualization to the world!
Thank You!