Xen and the Art of Virtualization

Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, Andrew Warfield

University of Cambridge Computer Laboratory

Kyle Schutt
CS 5204
Virtualization

• Abstraction of hardware resources
• Virtual Machine Monitors (Hypervisors)
• Key Players
  – Xen
  – VMWare
  – Hyper-V (Windows Server Virtualization)
  – KVM (Kernel Virtual Machine)
Virtualization Issues

• Isolation
• Reliability
• Security
• Scalability
• Performance
• Heterogeneous
Xen Virtualization

- Open source
- Paravirtualization and full virtualization
- Domain0 and DomainU
- Small footprint
- Direct hardware access
- Privilege control

Source: http://xen.org/
Overview

• Introduction
• Xen: Virtual Machine Monitor
• XenoLinux Evaluation
• Xen Current State
• Xen in Industry
• Xen Demo
• Discussion
Xen: Virtual Machine Monitor

- Hardware Layer
- x86 Paravirtualization
- Design Choices
  - Unmodified user application binaries
  - Full install of OSes
  - Paravirtualization – high performance and resource isolation
  - Transparent resource virtualization
Xen: Virtual Machine Monitor

Control Plane Software

User Software

User Software

User Software

GuestOS (Xenolinux)

GuestOS (Xenolinux)

GuestOS (XenBSD)

GuestOS (XenXP)

Xeno-Aware Device Drivers

Xeno-Aware Device Drivers

Xeno-Aware Device Drivers

Xeno-Aware Device Drivers

Domain0 control interface

virtual x86 CPU

virtual phy mem

virtual network

virtual blockdev

H/W (SMP x86, phy mem, enet, SCSI/IDE)
Xen: VMM Approach Overview

- x86 Specific Paravirtualization
- Data Transfers
- Intercommunication
- Porting Costs
- Control and Management
- Subsytems
Xen: VMM x86 Paravirtualization

• Memory Management
• CPU Scheduling
• Device I/O
Xen: VMM x86 Memory Management

• Registers allocations with Xen
• Untagged vs. Software-managed TLB

64MB Xen Reserved
Inaccessible by guest OS

Guest OS Page Table
Part of the guest OS memory allocation
Xen: VMM x86 CPU

- Privilege Levels
- Level 0
  - Typical OS
  - Xen Kernel
- Level 1
  - Guest OS w/ Xen
- Level 2
  - Unused
- Level 3
  - User Applications
Xen: VMM x86 Device I/O

• Paravirtualize Devices
  – Abstraction

• Mediator
  – Validation
  – Channel links

• I/O Rings
  – Shared memory
  – Descriptor rings
Xen: VMM Data Transfers

• I/O Rings

- Request Consumer
- Request Producer
- Response Producer
- Response Consumer

- **Request queue** - Descriptors queued by the VM but not yet accepted by Xen
- **Outstanding descriptors** - Descriptor slots awaiting a response from Xen
- **Response queue** - Descriptors returned by Xen in response to serviced requests
- **Unused descriptors**

Paul Barham, et al.
Xen: VMM Data Transfers

• I/O Rings

- Request Consumer
- Request Producer
- Response Producer
- Response Consumer

- **Request queue** - Descriptors queued by the VM but not yet accepted by Xen
- **Outstanding descriptors** - Descriptor slots awaiting a response from Xen
- **Response queue** - Descriptors returned by Xen in response to serviced requests
- **Unused descriptors**
Xen: VMM Data Transfers

• I/O Rings

- Request Consumer
- Request Producer
- Response Producer
- Response Consumer

Legend:
- **Request queue** - Descriptors queued by the VM but not yet accepted by Xen
- **Outstanding descriptors** - Descriptor slots awaiting a response from Xen
- **Response queue** - Descriptors returned by Xen in response to serviced requests
- **Unused descriptors**
Xen: VMM Data Transfers

• I/O Rings

Request Consumer

Request Producer

Response Producer

Response Consumer

- **Request queue** - Descriptors queued by the VM but not yet accepted by Xen
- **Outstanding descriptors** - Descriptor slots awaiting a response from Xen
- **Response queue** - Descriptors returned by Xen in response to serviced requests
- **Unused descriptors**
Xen: VMM Data Transfers

• I/O Rings

Request Consumer

Request Producer

Response Producer

Response Consumer

- **Request queue** - Descriptors queued by the VM but not yet accepted by Xen
- **Outstanding descriptors** - Descriptor slots awaiting a response from Xen
- **Response queue** - Descriptors returned by Xen in response to serviced requests
- **Unused descriptors**
Xen: VMM Intercommunication

- Hypercalls
  - Domain to Xen communication
  - Synchronous
  - Batched
- Events
  - Xen to Domain communication
  - Asynchronous
- Direct link through the hypervisor
Xen: VMM Pass-through

- New feature
- Performance increase
- Direct access to hardware resources
- No need for Domain0
Xen: Port Costs

- Idealized abstraction
- Linux and Windows
  - XenoLinux
- Paravirtualization port of x86 code base
- Device drivers
- Page-table entries
- Privileged subroutines
Xen: Control and Management

- Daemons
- XML RPC
- Xm
- Xend
- Libxenctrl
- Xenstored
- Qemu-dm
  - full virtualization daemon for disk/network I/O

Source: http://xen.org
Xen: VMM Subsystems
Xen: VMM Subsystems Overview

- CPU and Scheduling
- Timing
- Virtual Address Translation
- Physical Memory
- Device I/O
- Network
Xen: VMM CPU and Scheduling

• Privileges

• Scheduling
  – Borrowed Virtual Time
  – Low-latency
  – Favors new domains

• Exceptions
  – Guest OS registers handlers
  – Stack copied from guest OS
Xen: VMM Timing

• Real Time
  – Time since domain boot
  – Utilizes the clock speed of the processor
• Virtual Time
  – Execution time of the guest OS
• Wall-Clock Time
  – Current real time offset
• Timer Queues
  – Guest OS
Xen: VMM Virtual Addresses

• Page Tables
  – Guest OS allocates directly with Xen
  – Read-only
  – Updates are handle by hypercalls

• Validation
  – Manage page frame types
  – Reference counts
  – Updates based on types
Xen: VMM Virtual Addresses

- Frame Types
  - Page Directory
  - Page Table
  - Local Descriptor Table
  - Global Descriptor Table
  - Writable

- Batch updates in a single hypercall
Xen: VMM Physical Memory

• Reservations
• Balloon driver
  – Existing OS instructions
• Illusion of contiguous
• Mapping by guest OS
• Shared Translation Array
  – Accessible to all
  – Xen validated
Xen: VMM Device I/O

- Device abstractions
- Virtual Block Devices
  - Reordering
  - Uses I/O Ring
- Domain0
  - Disk
  - Network
- Round-robin scheduling
Xen: VMM Network Communication

- Asynchronous I/O Rings
  - Transmit
  - Receive
- Virtual Firewall-Router
- Virtual Network Interfaces
- Direct Memory Access
- Round-robin scheduling for packets
XenLinux Evaluation
XenoLinux Evaluation

• Comparison
  – VMWare Workstation (without ESX Server)
  – User-Mode Linux (UML)
  – Native Linux
  – XenoLinux (Linux 2.4.21)

• RedHat 7.2 distribution
XenoLinux Performance Evaluation

- SPEC INT2000
- Build Linux 2.4.21 with GCC 2.96
- Open Source Database Benchmark
  - Information Retrieval
  - On-Line Transaction Processing
- dbench
- SPEC WEB99
Figure 3: Relative performance of native Linux (L), XenoLinux (X), VMware workstation 3.2 (V) and User-Mode Linux (U).
XenoLinux Other Evaluations

- *lmbench* suite – microbenchmarks
  - 65%
  - Page Table Updates

- Network
  - *ttcp* benchmark
  - Negligible bandwidth differences
## XenoLinux Other Evaluations

<table>
<thead>
<tr>
<th>Config</th>
<th>null call</th>
<th>null I/O</th>
<th>openslct stat</th>
<th>sig closeTCP</th>
<th>sig inst</th>
<th>fork hndl</th>
<th>exec sh proc</th>
<th>proc proc</th>
</tr>
</thead>
<tbody>
<tr>
<td>L-SMP</td>
<td>0.53</td>
<td>0.81</td>
<td>2.10</td>
<td>3.51</td>
<td>23.2</td>
<td>0.83</td>
<td>2.94</td>
<td>143</td>
</tr>
<tr>
<td>L-UP</td>
<td>0.45</td>
<td>0.50</td>
<td>1.28</td>
<td>1.92</td>
<td>5.70</td>
<td>0.68</td>
<td>2.49</td>
<td>110</td>
</tr>
<tr>
<td>Xen</td>
<td>0.46</td>
<td>0.50</td>
<td>1.22</td>
<td>1.88</td>
<td>5.69</td>
<td>0.69</td>
<td>1.75</td>
<td>198</td>
</tr>
<tr>
<td>VMW</td>
<td>0.73</td>
<td>0.83</td>
<td>1.88</td>
<td>2.99</td>
<td>11.1</td>
<td>1.02</td>
<td>4.63</td>
<td>874</td>
</tr>
<tr>
<td>UML</td>
<td>24.7</td>
<td>25.1</td>
<td>36.1</td>
<td>62.8</td>
<td>39.9</td>
<td>26.0</td>
<td>46.0</td>
<td>21k</td>
</tr>
</tbody>
</table>

**Table 3: Imbench: Processes - times in \( \mu s \)**

<table>
<thead>
<tr>
<th></th>
<th>TCP MTU 1500</th>
<th></th>
<th>TCP MTU 500</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TX</td>
<td>RX</td>
<td>TX</td>
<td>RX</td>
</tr>
<tr>
<td>Linux</td>
<td>897</td>
<td>897</td>
<td>602</td>
<td>544</td>
</tr>
<tr>
<td>Xen</td>
<td>897 (-0%)</td>
<td>897 (-0%)</td>
<td>516 (-14%)</td>
<td>467 (-14%)</td>
</tr>
<tr>
<td>VMW</td>
<td>291 (-68%)</td>
<td>615 (-31%)</td>
<td>101 (-83%)</td>
<td>137 (-75%)</td>
</tr>
<tr>
<td>UML</td>
<td>165 (-82%)</td>
<td>203 (-77%)</td>
<td>61.1 (-90%)</td>
<td>91.4 (-83%)</td>
</tr>
</tbody>
</table>

**Table 6: ttcp: Bandwidth in Mb/s**
Further Evaluations

• Concurrency
  – SPEC WEB99

• Isolation
  – Fork Bomb
  – Intensive Disk Access

• Scalability
  – 1 to 128 domains
  – SPEC CINT2000
Figure 4: SPEC WEB99 for 1, 2, 4, 8 and 16 concurrent Apache servers: higher values are better.
Figure 6: Normalized aggregate performance of a subset of SPEC CINT2000 running concurrently on 1-128 domains
Xen Evaluation

- Isolation
- Reliability
- Security
- Scalability
- Performance
- Heterogeneous
Xen Current State

• Supported Architectures
  – x86
  – x86_64
  – PowerPC
  – IA64
  – ARM (in progress)

Xen Current State

• Host OSes
  – Ubuntu, CentOS, RedHat, etc.
  – Linux releases between 2009 and early 2011
    • Not in mainline kernel until 2.6.37
    • Some do not have domain0 support

Xen Current State

• Guest OSes
  – Patched Linux 2.6.23 with paravirtualization
  – OpenSolaris
  – Modified WindowsXP
  – Unmodified Windows
    • Intel VT-x
    • AMD-V

Xen in Industry

• Amazon Web Services
• Rackspace
• Other Commercial Applications
  – Citrix XenServer, XenDesktop, XenApp, XenClient
  – Oracle VM
  – Sun xVM

Sources: http://xen.org/, http://www.citrix.com
Xen Demo

• Recursive VMs
  – Win7 with VMWare Workstation 7.1.5
  – CentOS 5 with Xen 2.6
  – Fedora 7

• “Russian Doll Effect”
Discussions