Improving Agility and Elasticity in Bare-metal Clouds

Yushi Omote†, Takahiro Shinagawa‡, Kazuhiko Kato†
†University of Tsukuba, ‡The University of Tokyo
Bare-metal Clouds

An IaaS for high performance and device functionality

User

Provider

OS transparent

No Virtual Machine

Physical Machine
OS-deployment Problem

Long wait time sacrifices agility and elasticity

(1) Image Copy
   (Tens of minutes)

(2) Reboot from Local Disk
   (A few minutes)
Existing Approach 1
OS Streaming Deployment

[Clerc et al. IPCCC’10]
Existing Approach 2
Conventional VMMs

[VMware’01, Xen’03, KVM’07]

Streaming deployment with VMMs
- Agility and Elasticity
- OS transparency

Continuous virtualization overhead

✗ Performance
OS Deployment with a Special-purpose VMM

1) Streaming deployment
   - Agility and Elasticity
   - OS transparency

2) Seamless de-virtualization
   - Performance
Challenge
Expose & Control Physical Devices

Virtual Devices?
Direct I/O?

✗
✗

O
X

X
O
Device-interface-level I/O mediation

A device mediator performs:

1. **I/O interpretation** to understand I/O context
2. **I/O redirection** to perform network booting
3. **I/O multiplexing** to perform background install

Physical device interface
I/O Interpretation

Determine when/how to mediate I/O requests

OS → Device Driver → Device Mediator → VMM

Understand state transitions based on monitoring I/O

Device State Transitions
I/O Redirection

1. Interpret
2. Redirect
3. Restart

Image Server

VMM

Disk

Small Request

Interrupt

LBA=4
NUM=8

Data

LBA=4
NUM=8
I/O Multiplexing

1. Request

(1) Request

VMM Request

(2) Emulate

Idle State

(3) Queue

OS Request

Status Check

OS

Image Server

VMM

Disk

Server

VMM

Disk
CPU/Memory Virtualization for De-virtualizable VMM

**CPU**
- No indirection
- VMM runs passively with VMX
- No guest scheduling

**Memory**
- Guest Physical Address = VMM Physical Address
- Identity Mapping
  - VMM exposes physical memory
  - Mark VMM regions as *reserved* (via BIOS INT15/e802)
De-virtualization

(1) Turns off IO
VM exits

Find safe I/O timing

(2) Turns off
nested paging

Unsynchronized
TLB flush

(3) Turns off
CPU virtualization

Ease
VM exits condition
(VMXOFF Issue)
Performance Evaluation

- Deployed 32-GB OS Image (Ubuntu 14.04 64-bit)
  - OS-startup Time
  - Cassandra Throughput
  - Storage Throughput
  - InfiniBand Latency

A HPC Cluster

Intel Xeon X5680 (3.33 GHz) / 96GB RAM

<table>
<thead>
<tr>
<th>HDD 500GB/7200 RPM SATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mellanox InfiniBand (4X QDR)</td>
</tr>
<tr>
<td>Intel 82575 EM GbE Network Card</td>
</tr>
</tbody>
</table>

Interconnected by
A Mellanox Grid Director InfiniBand Switch &
A FUJITSU SR- S348TC1 GbE Switch
OS-startup Time

- **Image Copy**
- **Reboot+Firminit.**
- **VMM Boot**
- **OS Boot**

**Quick start up (8.6 times faster)**
Cassandra Throughput
(Throughout Deployment)

- Proposed
- KVM (No Background Install)

Eventual bare-metal performance
Seamless de-virtualization
Storage Throughput

<table>
<thead>
<tr>
<th></th>
<th>Throughput (MB/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bare-metal</td>
<td>117</td>
</tr>
<tr>
<td>Deploy</td>
<td>112</td>
</tr>
<tr>
<td>Devirt</td>
<td>112</td>
</tr>
<tr>
<td>KVM/Local</td>
<td>101</td>
</tr>
</tbody>
</table>

**Bare-metal performance**

- **Read**
  - Bare-metal: 117 MB/sec
  - Deploy: 112 MB/sec
  - Devirt: 112 MB/sec
  - KVM/Local: 101 MB/sec

- **Write**
  - Bare-metal: 112 MB/sec
  - Deploy: 112 MB/sec
  - Devirt: 115 MB/sec
  - KVM/Local: 100 MB/sec
InfiniBand RDMA latency

Latency (usec)

<table>
<thead>
<tr>
<th>Baremetal</th>
<th>Deploy</th>
<th>Devirt</th>
<th>KVM/Pass</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.30</td>
<td>1.30</td>
<td>1.30</td>
<td>1.61</td>
</tr>
</tbody>
</table>

Bare-metal performance
Conclusion

• Improved agility and elasticity in bare-metal clouds
  • De-virtualizable VMM with streaming deployment
    • Device-interface-level I/O mediation
  • Achieved quick startup of an OS
    • 8.6 times faster than image copy
    • Preserved high performance & OS-transparency
Future work

- Generating device mediators from specification
  - Reduce development cost of device mediators
- More advanced features of IaaS clouds
  - Live migration and checkpointing
Thank you