Agile Resource Management in a Virtualized Data Center

Wei Zhang\textsuperscript{1}, Hangwei Qian\textsuperscript{2}
Craig E. Wills\textsuperscript{1}, Michael Rabinovich\textsuperscript{2}

\textsuperscript{1}CS Department, Worcester Polytechnic Institute
\textsuperscript{2}EECS Department, Case Western Reserve University
Outline

- Background
- Approach
- Implementation
- Study & Results
- Future work & Summary
Virtualized Data Centers

Diagram showing a virtualized data center connected to the internet through DNS and switch 1. It consists of two data centers, one with a PM, VM, VM, VM, VM, and DB, and another with a PM, VM, VM, and DB. Users connect to the internet.
VM-based Resource Management

- **Previous work**
  - Power off/on a VM
  - Suspend/Resume a VM
  - Stop/Start application servers within a VM
  - Live VM migration

- **Our approach**
  - Ghost VMs
Motivation: Agility of previous work

- Power off/on a VM
  - Several minutes
- Suspend/Resume a VM
  - Several minutes (if including time of rejoining the application cluster)
- Stop/Start application servers within a VM
  - Several minutes
- Live VM migration
  - Tens of seconds (if including pre-copy phase)
What are Ghost VMs?

- “Invisible” VMs
  - to the content switch
- “Idle” VMs
  - to the PMs
- “Member” VMs
  - to the application cluster
- “Hot spare” VMs
  - to the application
Characteristics of Ghost VMs

- Negligible CPU and network usage
  - hide behind the switch (do not receive requests)
- Consume same amount memory as active VMs
- Agile (several seconds to become active)
  - No need to stop/start/suspend/resume
  - No need to rejoin application cluster
  - Only need to reconfigure the switch
- They are stepping stones to active VMs
Why ghost? why not just active?

- More active VMs ≠ better performance
  - Scheduling overhead, such as context switching
  - Minimal active VMs on each PM
- Extra capacity within a data center is not deployed to VMs until needed
  - Less overhead
  - Resources can be reassigned quickly
How does our algorithm work?

- **Make Decisions**
  - Capacity and Utilization
    - of PMs, VMs, and applications
    - Current and Projected

- **Enact Solutions**
  - Promote ghost VMs to active VMs
  - Demote active VMs to ghost VMs
  - Resume suspended VMs on disks
  - Suspend ghost VMs to disks
Implementation

Resource Pool

PM 1 2 ... n

Switch

Resource Manager

Resource Collector
- Get VM Usage
  - Reduce VM Capacity
  - Demote VMs

Resource Reallocator
- Get PM Usage
  - Increase VM Capacity
  - Promote VMs

Ghost Manager
- Monitor Ghost
- Prepare Ghost

Collect Resource
Prepare Ghost
Our Data Centers

- Two data centers
  - Located at WPI and CWRU
- Two types of virtualization
  - VMWare Server running on Debian Linux
  - VMWare ESX Server running on bare metal
- PMs and Switches
  - Intel 2-core with 2G RAM on 100M Net
  - Intel 4-core with 4G RAM on 100M Net
  - Nortel Alteon 2208
  - Cisco Content Switch 11501
- Applications and database
  - TPC-W bookstore on Websphere with Oracle
Results: Ghost Promotion

CPU Utilization (%)

Time (secs)

promotion called

new VM joined

load increased
Agility: Legacy vs Ghost

- **Approach**
  - Legacy vs Ghost

- **Load growth rate**
  - Fast vs Slow

- **Performance metrics**
  - Error (%)
  - Slow Responses (%)
  - Median Response Time (ms)
Agility: Legacy vs Ghost

• Results
  ◦ Ghost outperforms Legacy when load growth rate is fast

<table>
<thead>
<tr>
<th>Approach / Growth Rate</th>
<th>% Slow Responses (&gt; 500ms)</th>
<th>Median Response Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ghost / Fast</td>
<td>2.6</td>
<td>46</td>
</tr>
<tr>
<td>Legacy / Fast</td>
<td>5.7</td>
<td>69</td>
</tr>
<tr>
<td>Ghost / Slow</td>
<td>3.4</td>
<td>38</td>
</tr>
<tr>
<td>Legacy / Slow</td>
<td>3.4</td>
<td>37</td>
</tr>
</tbody>
</table>

Legacy vs Ghost in our VMWare ESX data center
Performance: Fixed vs Manual vs Ghost

- Approach
  - Fixed
  - Manual

- Multiple applications workload

![Graph showing client load (EBs) over time for different applications.](image)
Performance:

Fixed vs Manual vs Ghost

- **Mean number of active VMs**
  - Achieve similar performance with less number of active VMs on average

- **Parameters**
  - High watermark (HW)
  - Low watermark (LW)
Future Work

- Global resource management
  - Balance load between data centers
  - Geographically distributed data centers
- Scalability for mega data centers
  - Scalability of our approach
- Other types of resource
  - Memory, network, disk, etc.
Summary

- Web applications introduce resource provisioning challenge
  - Virtualization is promising in utility computing
  - Agility is important to data centers
  - Previous approaches have advantages and disadvantages

- We developed, implemented and tested a virtualized data center solution
  - Use Ghost VMs
  - Achieve better agility
Thank you!

Questions?