



Agile Resource Management in a Virtualized Data Center

Wei Zhang¹, Hangwei Qian²

Craig E. Wills¹, Michael Rabinovich²

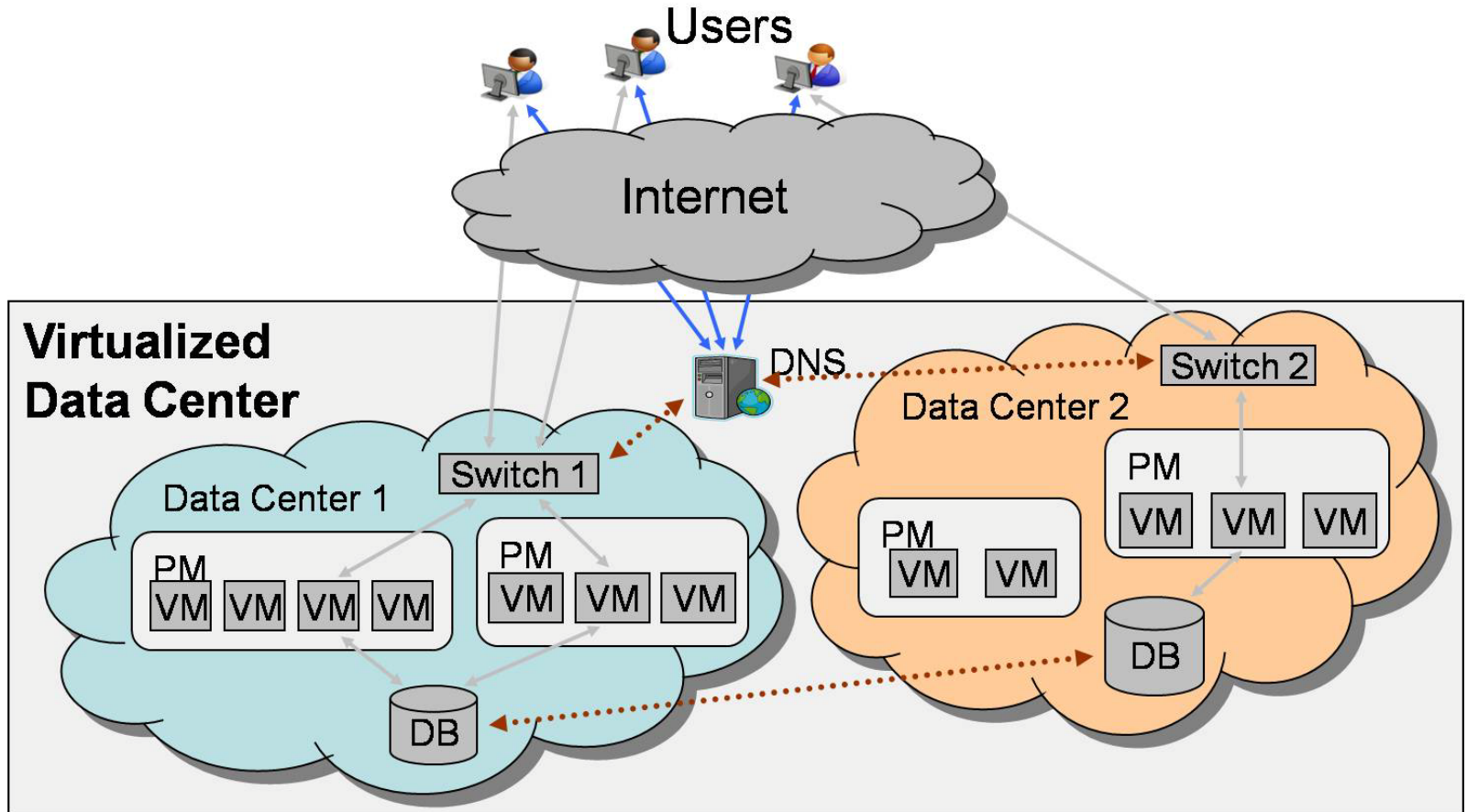
¹CS Department, Worcester Polytechnic Institute

²EECS Department, Case Western Reserve University

Outline

- Background
- Approach
- Implementation
- Study & Results
- Future work & Summary

Virtualized Data Centers



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

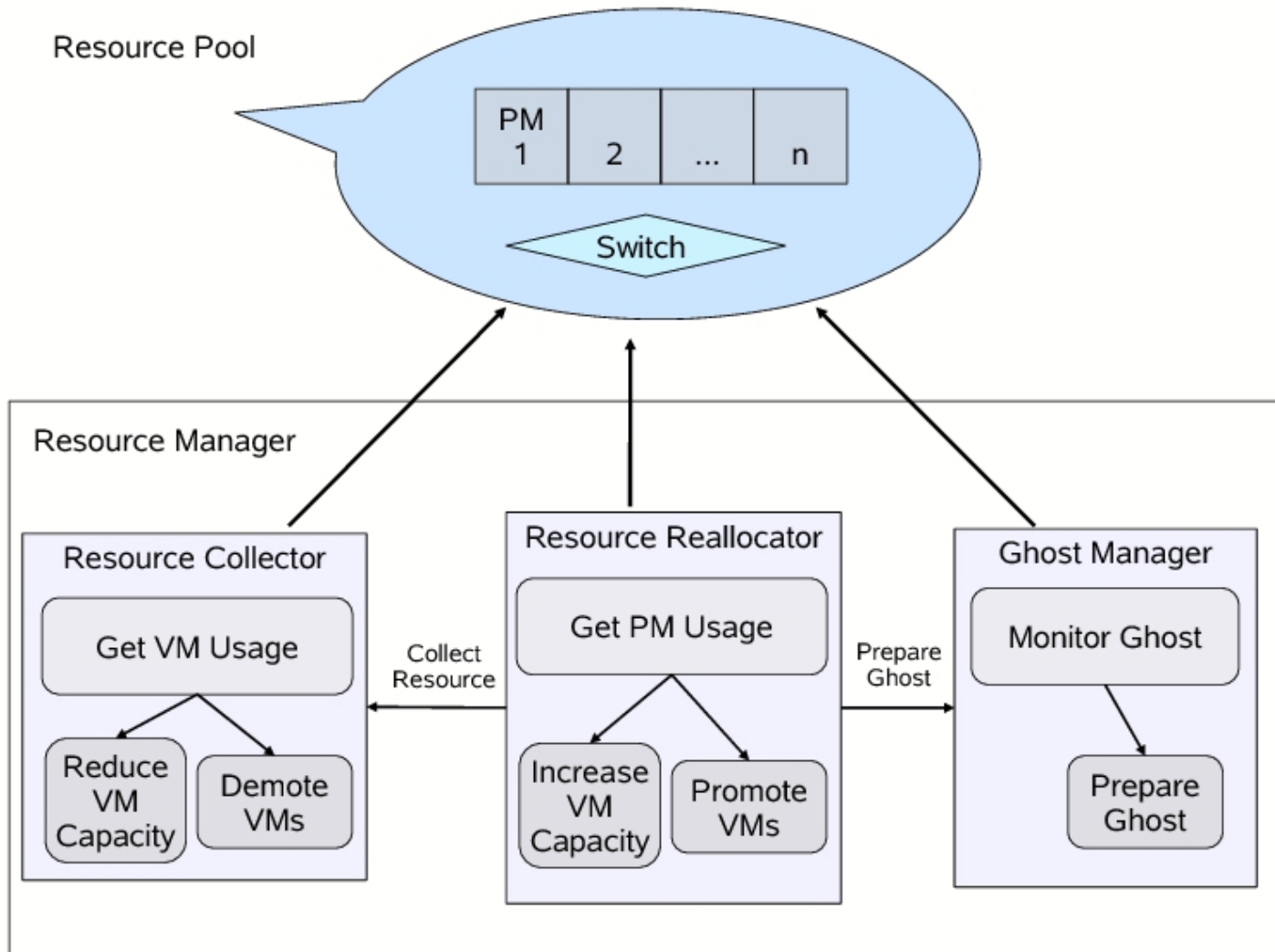
Why ghost? why not just active?

- More active VMs \neq 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

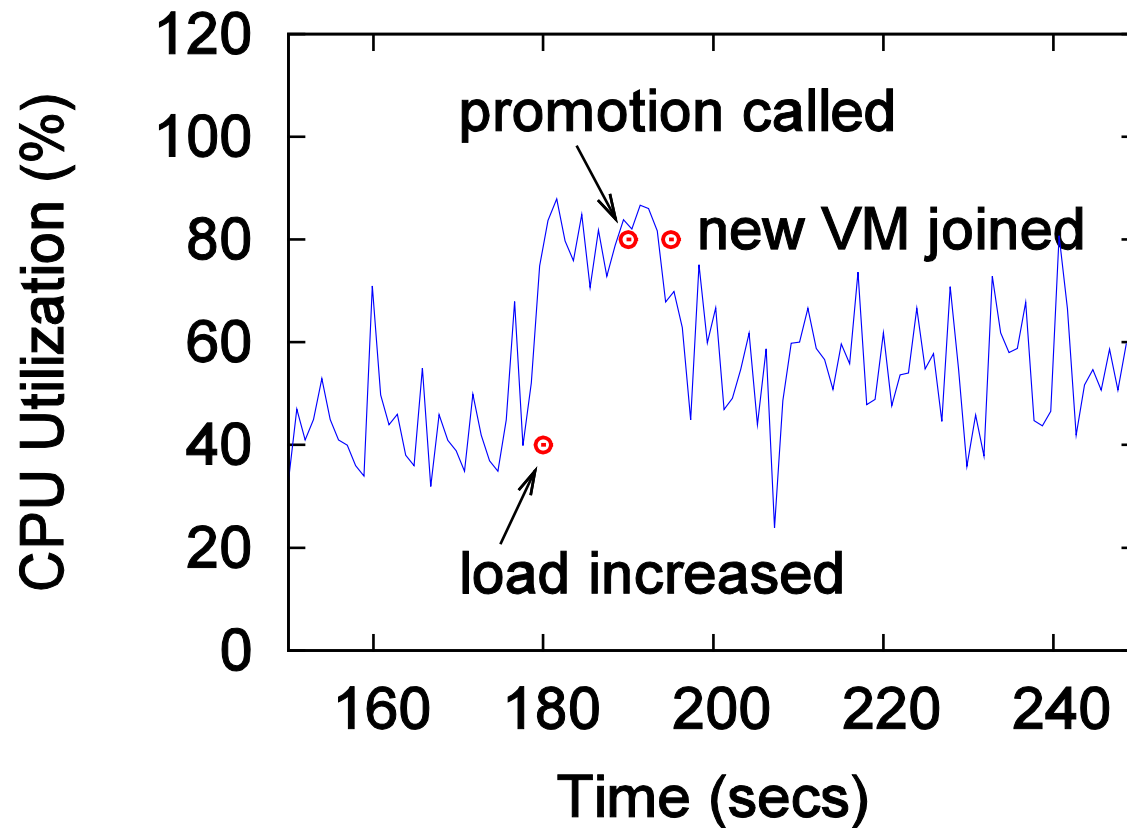
Implementation



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



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

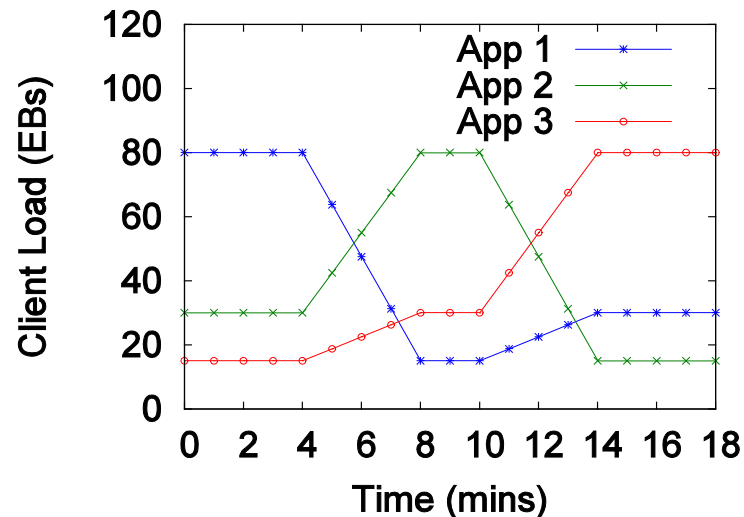
Legacy vs Ghost in our VMWare ESX data center

| Approach / Growth Rate | % Slow Responses (> 500ms) | Median Response Time (ms) |
|------------------------|----------------------------|---------------------------|
| Ghost / Fast | 2.6 | 46 |
| Legacy / Fast | 5.7 | 69 |
| Ghost / Slow | 3.4 | 38 |
| Legacy / Slow | 3.4 | 37 |

Performance:

Fixed vs Manual vs Ghost

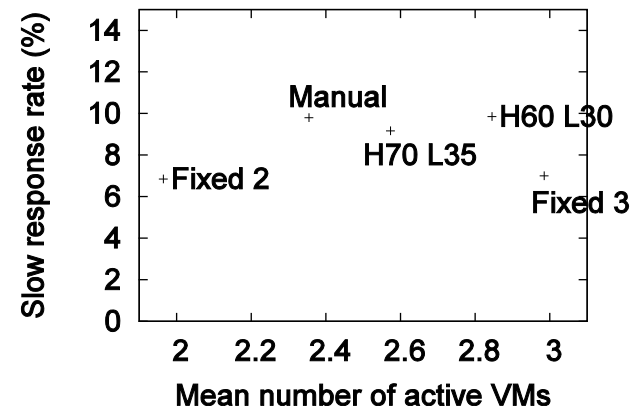
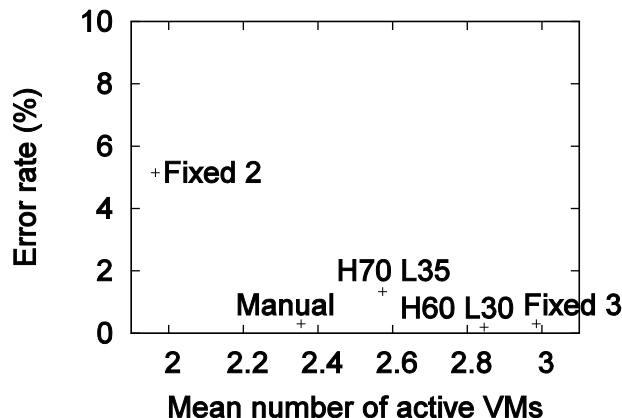
- Approach
 - Fixed
 - Manual
- Multiple applications workload



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?