Performance Evaluation of Virtualization Technologies

Saad Arif

Dept. of Electrical Engineering and Computer Science
University of Central Florida - Orlando, FL

September 19, 2013
1 Introduction
1 Introduction

2 Xen and OpenVZ Systems
1. Introduction

2. Xen and OpenVZ Systems

3. Testbed Architecture
1 Introduction

2 Xen and OpenVZ Systems

3 Testbed Architecture

4 Experimental Results
1 Introduction

2 Xen and OpenVZ Systems

3 Testbed Architecture

4 Experimental Results

5 Conclusion
Introduction

Xen and OpenVZ Systems

Testbed Architecture

Experimental Results

Conclusion
What we have already covered ...

- Virtualization.
- Layering and virtualization.
- Virtual machine monitor.
- Virtual machine.
- Performance and security isolation.
- Architectural support for virtualization.
- x86 support for virtualization.
- Full and paravirtualization.
- Xen 1.0 and Xen 2.0.
- Performance comparison of virtual machine monitors.
- The darker side of virtualization.
- Software fault isolation.
Overview

• Server consolidation
  • Reduces cost
  • Improves efficiency

• New challenges
  • Choosing the right virtualization technology
  • Consolidation configuration

• Performance evaluation of two representative virtualization technologies
  • Xen system
  • OpenVZ system
Overview (contd)

- Comparison in terms of
  - Application performance
  - Resource consumption
  - Scalability
  - Low-level system metrics like
    - Cache misses
    - Virtualization-specific metrics
What is Virtualization?

"The ability to run multiple operating systems on a single physical system and share the underlying hardware resources"

VMware white paper, Virtualization Overview
Motivation

- Rapid growth of servers in data centers
- High maintenance costs
- Enterprise data centers under-utilized
- Typical average utilization \( \leq 30\% \)
- Overload under peak demands
- Resulting in
  - Lower application throughput
  - Longer latency
- Significant interest in Virtualization

Figure: Example peak/off peak CPU utilization
Types of Virtualizations

- Hypervisor-based technology
  - VMware
  - Microsoft Virtual Server
  - Xen

- Operating system (OS) level virtualization
  - OpenVZ
  - Linux VServer
  - Solaris Zones

Both allow a single physical server to be partitioned into multiple isolated virtual machines (VMs) for running multiple applications at the same time

- Easier centralized server administration
- Higher operational efficiency
Capacity Management

- Enterprise applications often have resource demands that vary over time
- If consolidated into VMs on a shared server
  - Dynamic resource allocation
  - Maintained QoS
  - Efficient server resource utilization

(a) CPU consumption of node 1

(b) CPU consumption of node 2

(c) Sum of CPU consumptions from both nodes
1. Introduction

2. Xen and OpenVZ Systems

3. Testbed Architecture

4. Experimental Results

5. Conclusion
Xen System

- Paravirtualization technology
- Linux based
- Allows multiple guest operating systems
- Provides
  - Thin software virtualization layer between guest OS and HW
  - Presents hardware abstraction
  - CPU scheduler
OpenVZ System

- Linux based
- Allows creation of secure, isolated virtual environments (VEs)
- Each VE
  - Behaves as a stand-alone server
  - Can be rebooted independently
  - Can be setup with a different distribution with separate root directory
- Single kernel shared by all VEs
- Low level of fault isolation compared to Xen
1 Introduction

2 Xen and OpenVZ Systems

3 Testbed Architecture

4 Experimental Results

5 Conclusion
Performance Evaluation Critereon

- How is application-level performance, including throughput and response time, impacted?
- How does application-level performance scale up as workload increases?
- How is application-level performance affected when multiple tiers of each application are placed on virtualized servers in different ways?
- As the number of multi-tiered applications increases, how do application-level performance and resource consumption scale?
Setup

- **Base system**
  - Vanilla Linux 2.6 kernel

- **Xen system**
  - Xen 3.0.3

- **OpenVZ system**
  - ovz-stable-2.6 kernel

- **Virtual Machine**
  - One or more virtual VMs supported by either Xen or OpenVZ
  - Sensors collecting
    - CPU consumption
    - Memory consumption
    - Response time
Setup (contd)

(a) Testbed setup

(b) A virtualized server
Oprofile

- Tool to measure hardware events
- Requires hardware performance counters
- Specific Counters:
  - CPU_CLK_UNHALT - number of CPU cycles outside halt state
  - RETIRED_INSTRUCTIONS - number of retired instructions
  - L2_CACHE_MISS - number of L2 cache misses and main memory hits
Experiment Design

- Benchmark
  - RUBiS: online auction site
    - Multi-tiered application
    - Web-tier: runs Apache web server with PHP
    - DB-tier: runs MySQL database server

- Multiple clients connect to Web-tier

- Client sessions
  - Connect
  - Browse
  - Buy/sell

- Client waits for each request completion before starting a new one (closed-loop behaviour)
Load distribution: browsing mix
- Higher load on Web-tier than DB-tier

Goal
- Quantitatively evaluate impact of virtualization
- Evaluate application-level performance including
  - Throughput
  - Response time
Experiment Configurations

- Single-node: Both Web and DB tiers in a single node
- Two-node: Web and DB tiers in separate nodes
- Two-node (scaled): Multiple web/DB tiers in each nodes
Single-Node

- Both Web and DB tiers hosted on single node
- Xen/OpenVZ run two tiers in separate VM
- 500 - 800 concurrent clients
Performance Evaluation: Single-Node

- Similar performance of base, Xen and OpenVZ
- Throughput increases linearly with number of threads

Figure: Throughput
Performance Evaluation: Single-Node (contd)

- Higher performance overhead in Xen compared to Open VZ
- Base and OpenVZ show slight increase
- Xen shows growth of 600% (from 18 ms to 130 ms)
- At 800 threads, Xen is over 4 times of OpenVZ

Figure: Average response time

Xen system is less scalable with the workload than Open VZ or a nonvirtualized system
Performance Evaluation: Single-Node (contd)

- Average CPU consumption of Web and DB tiers as a function of workload
- Goes up linearly with number of threads for both tiers
- DB consumption low in browsing mix (1 - 4% of total CPU) hence not marked difference
- Web tier for Xen twice that of base
- OpenVZ stays close to base
- Xen shows higher slope of increase

Xen system shows higher CPU overhead, related to the higher response time
Oprofile Analysis: Single-Node

- Aggregate values of counters at 800 threads
- For OpenVZ, counter values for whole system
- For Xen, counters for each domain
- DomU = sum of values from Web and DB domains
- All counters normalized with base
Oprofile Analysis: Single-Node (contd)

- OpenVZ shows counter less than twice of base
- L2 cache misses in Xen 11 times of base

Xen’s higher CPU overhead and response time due to high L2 cache misses
Two-Node

- Web and DB tier of one RUBiS application on separate nodes
- Xen/OpenVZ run one VM on each node (Web or DB)
- 500 - 800 concurrent clients

We will omit throughput comparison for two-node as it is similar to the single-node
Performance Evaluation: Two-Node

- Small overhead in Open VZ compared to base case
- Xen shows increase of 115% (13 ms to 28 ms)

Figure: Average response time

Xen’s increase in response time is significantly lower than Xen in single-node case
Performance Evaluation: Two-Node (contd)

- Average CPU consumption of Web and DB tiers as a function of workload
- Similar trend as single-node case
- Goes up linearly with number of threads for both tiers
- Web tier consumption higher than DB
- Xen shows higher slope of increase for Web tier

**Figure**: Average CPU consumption
Performance Evaluation: Two-Node (contd)

- Dom0 CPU consumptions of single-node vs two-node configurations for Xen
- For two-node case, Dom0 = sum of Dom0 for both nodes
- Dom0 CPU consumption remains low (below 4%)
- Shows slow linear increase

Dom0 CPU consumption is mostly workload dependent, and there is very little fixed cost.

Figure: Single-node vs two-node
Oprofile Analysis: Two-Node

- OpenVZ L2 cache misses less than twice of the base
- L2 cache misses in Xen 5 - 10 times of base

**Figure: Web tier**

Xen’s higher CPU overhead and response time due to high L2 cache misses
Oprofile Analysis: Two-Node (contd)

- OpenVZ L2 cache misses about 4 times of the base
- L2 cache misses in Xen 5 - 10 times of base

Figure: DB tier
Oprofile Analysis: Two-Node (contd)

- For two-node, each counter = sum of counter from each node
- L2 cache misses are higher for single-node compared to two-node for both Xen and OpenVZ
  - Due to extra overhead caused by two VMs on one node

Figure: Single-node vs two-node
Scalability Evaluation

- Increase the number of RUBiS instances on two nodes
  - One to two
  - Two to four
- Compare the scalability of Xen and OpenVZ
  - Application performance
  - Resource consumption
Scalability Evaluation (contd)

- Remains relatively constant for OpenVZ.
- Goes up about 500% (15 ms to roughly 90 ms) for Xen

**Figure**: Two-node two instances - average response time
Remains below 30 ms for OpenVZ.
Goes up about 600% (20 ms to between 140 and 200 ms) for Xen

**Figure**: Two-node four instances - average response time

Average response time in Xen increases with increasing number of instances
Scalability Evaluation (contd)

- **Xen**
  - For two-node case
    - One instance: grows to 28 ms
    - Four instances: grows to 158 ms
    - Overall 400% increase!

- **OpenVZ system**
  - Increase is only about 100%

**Figure**: Average response time comparison - All configurations at 800 threads
Scalability Evaluation (contd)

- **Xen**
  - Roughly twice of OpenVZ
  - Already becoming overloaded with four instances
- **OpenVZ system**
  - Total consumption below 60%
  - Can fit at least two more instances of RUBiS

**Figure**: Web tier CPU consumption comparison - All configurations at 800 threads
1 Introduction

2 Xen and OpenVZ Systems

3 Testbed Architecture

4 Experimental Results

5 Conclusion
Conclusion

- Xen shows higher virtualization overhead than Open VZ
- Performance degradation in Xen increases as application workloads increase
- Virtualization overhead observed in Open VZ is limited
- Web tier CPU consumption for Xen is roughly twice that of the base system or Open VZ
- Main reason of performance overhead in Xen is large number of L2 cache misses
Questions