CUDA in the Cloud – Enabling HPC Workloads in OpenStack

John Paul Walters
Computer Scientist, USC Information Sciences Institute
jwalters@isi.edu

With special thanks to Andrew Younge (Indiana Univ.) and Massimo Bernaschi (IAC-CNR)
Outline

- Introduction
- OpenStack Background
- Supporting Heterogeneity
- Supporting GPUs
- Performance Results
- Current Status
- Future Work
Introduction

- Cloud computing is traditionally seen as a resource for IT
  - Web servers, databases
- More recently researchers have begun to leverage the public cloud as an HPC resource
  - AWS virtual cluster is 102 on Top500 list
- Major difference between HPC and IT in the cloud:
  - Types of resources, heterogeneity
- Our contribution: we’re developing the heterogeneous HPC extensions for the OpenStack cloud computing platform
Why A Heterogeneous Cloud?

- Typical cloud infrastructures focus on commodity hardware.
  - Choose number of CPUs, memory, and disk
  - Often not a good match for technical computing
- Advantage of Heterogeneity
  - Leverage GPUs and other accelerators
  - Power efficiency, performance
  - Process technical workloads
- Move away from batch/grid schedulers
  - Give users access to customizable instances
  - Dedicated virtual machine
Benefits of Heterogeneity

CPU: $10^8$ samples  GPU: $10^{10}$ samples

136.2 seconds  139.5 seconds

Each architecture performs well on different applications.
OpenStack Background

- OpenStack founded by Rackspace and NASA
- In use by Rackspace, HP, and others for their public clouds
- Open source with hundreds of participating companies
- In use for both public and private clouds
- Current stable release: OpenStack Folsom
  - OpenStack Grizzly to be released in April
OpenStack Architecture

Image Source: Ken Peppe
(http://ken.pepple.info/openstack/2012/09/25/openstack-folsom-architecture/)
Supporting Heterogeneity

• Current cloud services have limited resource options
  • M1.large = 7.5 GB RAM, 2 virtual cores, 850 GB disk
  • x86 and x86_64 only
• Users should be able to customize instance types to needs
  • E.g. 16 GB RAM, 4 virtual cores, SSE4.2, 2xKepler GPUs
  • Hypervisor type
Supporting GPUs

- We’re pursuing two approaches for GPU support in OpenStack
  - LXC support for container-based VMs
    - Good for non-virtualization-friendly GPUs
    - Near-native performance
    - Can’t run non-Linux guests
  - Xen support for fully virtualized guests
    - Fully virtualized guests, can run Windows, etc.
    - Some overhead, especially for PCIe transfers

- Our OpenStack work currently supports GPU-enabled LXC containers
  - Xen is in development, preliminary results shown today
Integrating GPU Support into OpenStack

- Our GPU work is based off of the OpenStack’s libvirt module
  - Libvirt supports KVM/QEMU, LXC
  - Libvirt-based Xen support is experimental

- After instances boot, LXC’s GPUs are whitelisted and their major/minor device IDs are passed into the VM

- GPU Instances are launched just like any other:

  ```
  Euca-run-instances -k <my key> -t cg1.small <machine image>
  ```

  cg1 instance types represent GPUs
Test Results: LXC and Xen

- 3 data sets (all using CUDA 5)
  - NVIDIA CUDA samples
  - SHOC multi-GPU
  - BFS/Graph500 multi-GPU

Hardware

LXC tests
- 2x Xeon E5520
- 96 GB RAM
- 1x Tesla S2050 (4 GPUs)
- RHEL 6.1
  - 2.6.38.2 kernel (non-stock)

Xen Tests
- 2x Xeon X5660
- 192 GB RAM
- 2x NVIDIA Tesla C2075
- Centos 6.3 (with Xen 4.1.2)
  - 2.6.32-279 kernel (guest, stock)
LXC Bandwidth Results

LXC vs. Base, Host to Device Bandwidth, Pinned

LXC vs. Base, Device to Host Bandwidth, Pinned

LXC mirrors base bandwidth
LXC SHOC Results

LXC SHOC Benchmarks, 1 Node 4 GPUs. Performance Relative to Base

Relative Performance

Higher is Better
LXC Graph500 Results

BFS/Graph500 2,4 GPUs, 1 Node

Problem Size

TEPS: Traversed Edges per Second
Xen CUDA 5 Samples

Xen CUDA Samples, Performance Relative to Base

Higher is Better
Xen Bandwidth Results

Xen vs. Base, Host to Device
Bandwidth, Pinned

Bandwidth, MB/s

0 1000 2000 3000 4000 5000 6000 7000

Data Size, KB

1 4 16 64 256 1024 4096 16384 65536

Xen vs. Base, Device to Host
Bandwidth, Pinned

Bandwidth, MB/s

0 1000 2000 3000 4000 5000 6000

Data Size, KB

1 4 16 64 256 1024 4096 16384 65536

Base
Xen VM
Xen SHOC Benchmarks

Xen SHOC Benchmarks, 1 Node 2 GPUs. Performance Relative to Base

Higher is Better
Xen Graph500 Results

![BFS/Graph500, 2 GPUs, 1 Node](image)

**TEPS**: Traversed Edges per Second
Current Status

- Source code is available now
  - [https://github.com/usc-isi/nova](https://github.com/usc-isi/nova)
- Includes support for heterogeneity
  - GPU-enabled LXC instances
  - Bare-metal provisioning
  - Architecture-aware scheduler
- We’re working towards integrating additional support into the OpenStack Grizzly and H-releases
Future Work

- Add support for high speed networking as a resource
  - Infiniband via SR-IOV
  - Enable features like GPUdirect
- Add support for the Kepler architecture
- Merge the Xen support work into Upstream OpenStack