Building Clouds using Commodity, Open-Source Software Components

Rich Wolski

Chris Grzegorczyk, Dan Nurmi, Graziano Obertelli, Woody Rollins, Sunil Soman, Lamia Youseff, Dmitrii Zagorodnov

Computer Science Department

University of California, Santa Barbara

UCSB

Eucalyptus

Exciting Weather Forecasts

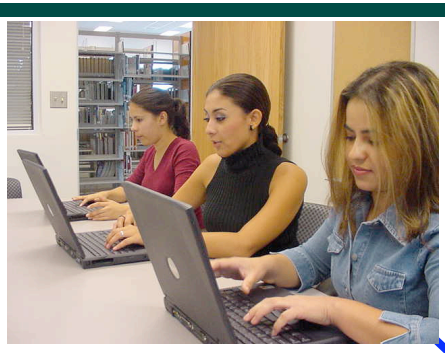
UCSB

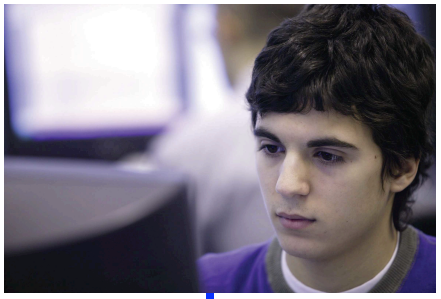
Eucalyptus

Commercial Cloud Formation

Amazon Elastic Compute Cloud (Amazon EC2) - *Beta*

What is a Cloud?





SLAs

Web Services

Virtualization

UCSB

Eucalyptus

Public Clouds (Now)

- Large scale infrastructure available on a rental basis
  - Operating System virtualization (e.g. Xen) provides CPU isolation
  - "Roll-your-own" network provisioning provides network isolation
  - Locally specific storage abstractions
- Fully customer self-service
  - Service Level Agreements (SLAs) are advertized
  - Requests are accepted and resources granted via web services
  - Customers access resources remotely via the Internet
- Accountability is e-commerce based
  - Web-based transaction
  - "Pay-as-you-go" and flat-rate subscription
  - Customer service, refunds, etc.

How do they work?

- Public clouds are opaque
  - What applications will work well in a cloud?
- Many of the advantages offered by Public Clouds appear useful for "on premise" IT
  - Self-service provisioning
  - Legacy support
  - Flexible resource allocation
- What extensions or modifications are required to support a wider variety of services and applications?
  - Data assimilation
  - Multiplayer gaming
  - Mobile devices

Open Source Cloud Infrastructure

- **Simple**
  - Transparent => need to "see" into the cloud
  - Scalable => complexity often limits scalability
- **Extensible**
  - New application classes and service classes may require new features
  - Clouds are new => need to extend while retaining useful features
- **Commodity-based**
  - Must leverage extensive catalog of open source software offerings
  - New, unstable, and unsupported infrastructure design is a barrier to uptake, experimentation, and adoption
- **Easy**
  - To install => system administration time is expensive
  - To maintain => system administration time is really expensive

On a Clear Day…

- **Globus/Nimbus**
  - Client-side cloud-computing interface to Globus-enabled TeraPort cluster at U of C
  - Based on GT4 and the Globus Virtual Workspace Service
  - Shares upsides and downsides of Globus-based grid technologies
- **Enomalism (now called ECP)**
  - Start-up company distributing open source
  - REST APIs
- **Reservoir**
  - European open cloud project
  - Many layers of cloud services and tools
  - Ambitious and wide-reaching but not yet accessible as an implementation

UCSB

Eucalyptus

- Elastic Utility Computing Architecture Linking Your Programs To Useful Systems
- Web services based implementation of elastic/utility/cloud computing infrastructure
  - —Linux image hosting ala Amazon
- *How do we know if it is a cloud?*
  - —Try and emulate an existing cloud: Amazon AWS
- Functions as a software overlay
  - —Existing installation should not be violated (too much)
- Focus on installation and maintenance
  - —*"System Administrators are people too."*

Goals for Eucalyptus

- Foster greater understanding and uptake of cloud computing
  - Provide a vehicle for extending what is known about the utility model of computing
- Experimentation vehicle prior to buying commercial services
  - Provide development, debugging, and "tech preview" platform for Public Clouds
- Homogenize local IT environment with Public Clouds
  - AWS functionality locally makes moving using Amazon AWS easier, cheaper, and more sustainable
- Provide a basic software development platform for the open source community
  - E.g. the "Linux Experience"
- Not a designed as a replacement technology for AWS or any other Public Cloud service

Open Source Cloud Anatomy

- Extensibility
  - Simple architecture and open internal APIs
- Client-side interface
  - Amazon's AWS interface and functionality (familiar and testable)
- Networking
  - Virtual private network per cloud
  - Must function as an overlay => cannot supplant local networking
- Security
  - Must be compatible with local security policies
- Packaging, installation, maintenance
  - system administration staff is an important constituency for uptake

Notes from the Open Source Cloud

- Private clouds are really hybrid clouds
  - Users want private clouds to export the same APIs as the public clouds
- In the Enterprise, the storage model is key
  - Scalable "blob" storage doesn't quite fit the notion of "data file."
- Cloud Federation is a policy mediation problem
  - No good way to translate SLAs in a cloud allocation chain
  - "Cloud Bursting" will only work if SLAs are congruent
- Customer SLAs allow applications to consider cost as first-class principle
  - Buy the computational, network, and storage capabilities that are required

Cloud Mythologies

- Cloud computing infrastructure is just a web service interface to operating system virtualization.
  - "I'm running Xen in my data center – I'm running a private cloud."
- Cloud computing imposes a significant performance penalty over "bare metal" provisioning.
  - "I won't be able to run a private cloud because my users will not tolerate the performance hit."
- Clouds and Grids are equivalent
  - "In the mid 1990s, the term grid was coined to describe technologies that would allow consumers to obtain computing power on demand."

Clouds and Virtualization

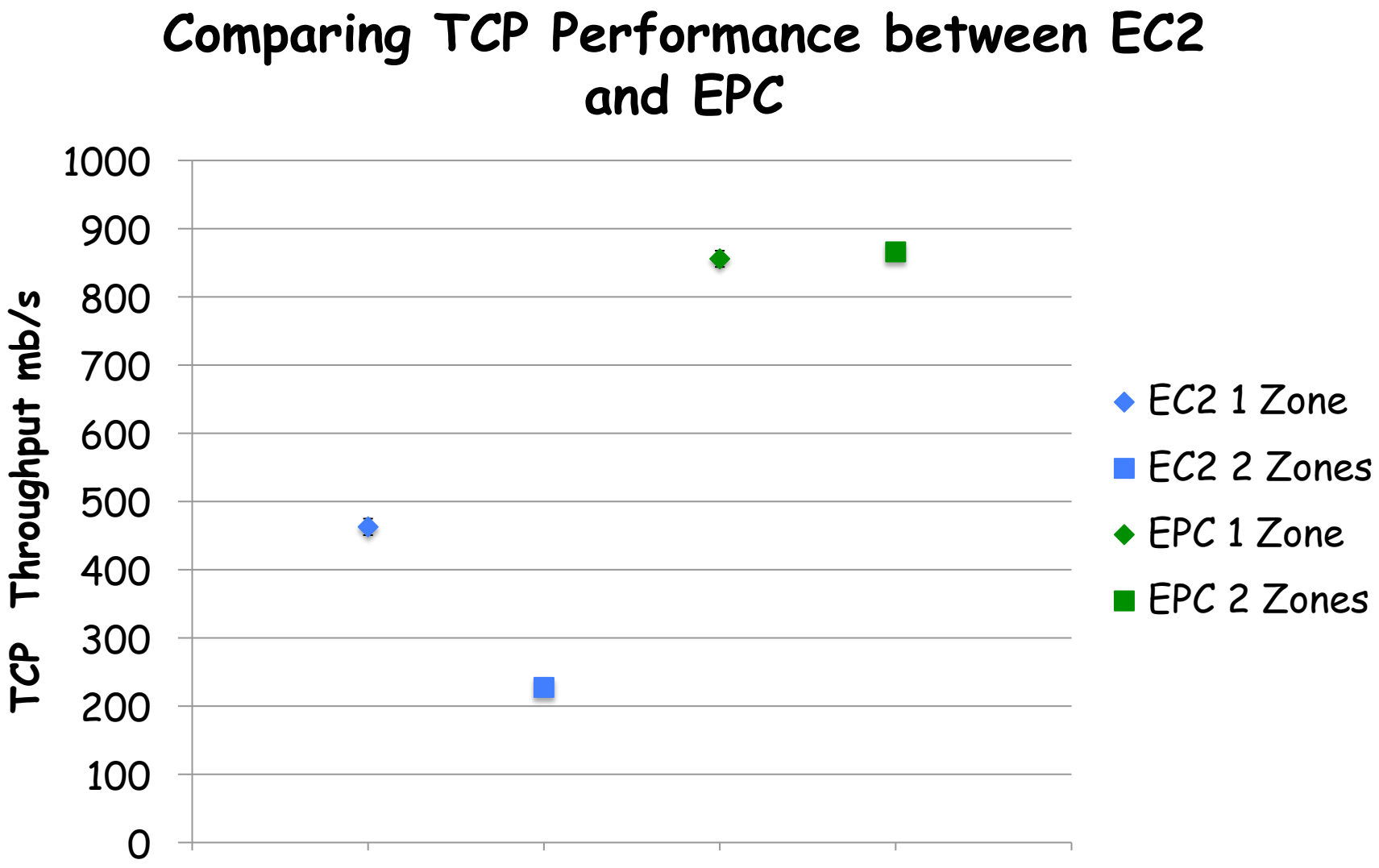
- Operating System virtualization (Xen, KVM, VMWare, HyperV) is only apparent for IaaS
  - AppEngine = BigTable + MapReduce
- Hypervisors virtualize CPU, Memory, and local device access as a single virtual machine (VM)
- IaaS Cloud allocation is
  - Set of VMs
  - Set of storage resources
  - Private network
- Allocation is atomic

Requires more than
A set of Hypervisors

Cloud Speed

- Extensive performance study using HPC applications and benchmarks
- Two questions:
  - What is the performance impact of virtualization?
  - What is the performance impact of cloud infrastructure?
- Tested Xen, Eucalyptus, and AWS (small SLA)
- Many answers:
  - Random access disk is slower with Xen
  - CPU bound can be *faster* with Xen -> depends on configuration
  - Kernel version is far more important
  - Eucalyptus imposes no statistically detectable overhead
  - AWS small appears to throttle network bandwidth and (maybe) disk bandwidth -> *$0.10 / CPU hour*

Gratuitous Performance Slide



UCSB

Eucalyptus

Clouds Versus Grids

- Rich's assertion: Clouds and Grids are distinct
- Cloud
  - Full private cluster is provisioned
  - Individual user can only get a tiny fraction of the total resource pool
  - No support for cloud federation except through the client interface
  - Opaque with respect to resources
- Grid
  - Built so that individual users can get most, if not all of the resources in a single request
  - Middleware approach takes federation as a first principle
  - Resources are exposed, often as bare metal
- These differences mandate different architectures for each

Open Source Cloud Ecosystem

- **AppScale**
  - Google App Engine inside EC2/Eucalyptus
  - Multiple scalable database back ends
    - http://appscale.cs.ucsb.edu

- **Rightscale**
  - Local enterprise focused on providing client tools as SaaS hosed in AWS
  - "Turing Test" for Eucalyptus
    - Can Rightscale "tell" that it isn't talking to EC2?
  - Uses the REST interface
  - Available for EPC
    - http://eucalyptus.rightscale.com
  - Next release any Eucalyptus cloud will be able to register with a free RightScale image

Our Roadmap

- 5/28/08 – Release 1.0 shipped
- 8/28/08 – EC2 API and initial installation model in V1.3
  - Completes overlay version
- 12/16/08 – Security groups, Elastic IPs, AMI, S3 in V1.4

- 4/01/09 – EBS, Metadata service in V1.5
- 4/23/09 - Ubuntu release
- 5/15/09 – Final feature release as V1.6
  - Completes AWS specification as of 1/9/2009
- 6/15/09 – Final bug-fix release
  - "core" opens for community contributions

Thanks and More Information

- National Science Foundation
  - VGrADS Project
- SDSC, CNSI, IU, Rice University
- RightScale.com
- The Eucalyptus Development Team at UCSB is
  - Chris Grzegorczyk -- grze@cs.ucsb.edu
  - Dan Nurmi -- nurmi@cs.ucsb.edu
  - Graziano Obertelli -- graziano@cs.ucsb.edu
  - Sunil Soman -- sunils@cs.ucsb.edu
  - Lamia Youseff -- lyouseff@cs.ucsb.edu
  - Dmitrii Zagordnov -- dmitrii@cs.ucsb.edu
- rich@cs.ucsb.edu
- http://eucalyptus.cs.ucsb.edu