

# Using GENI in Large Classes within a Cloud Computing course

**Prasad Calyam, Ph.D.**

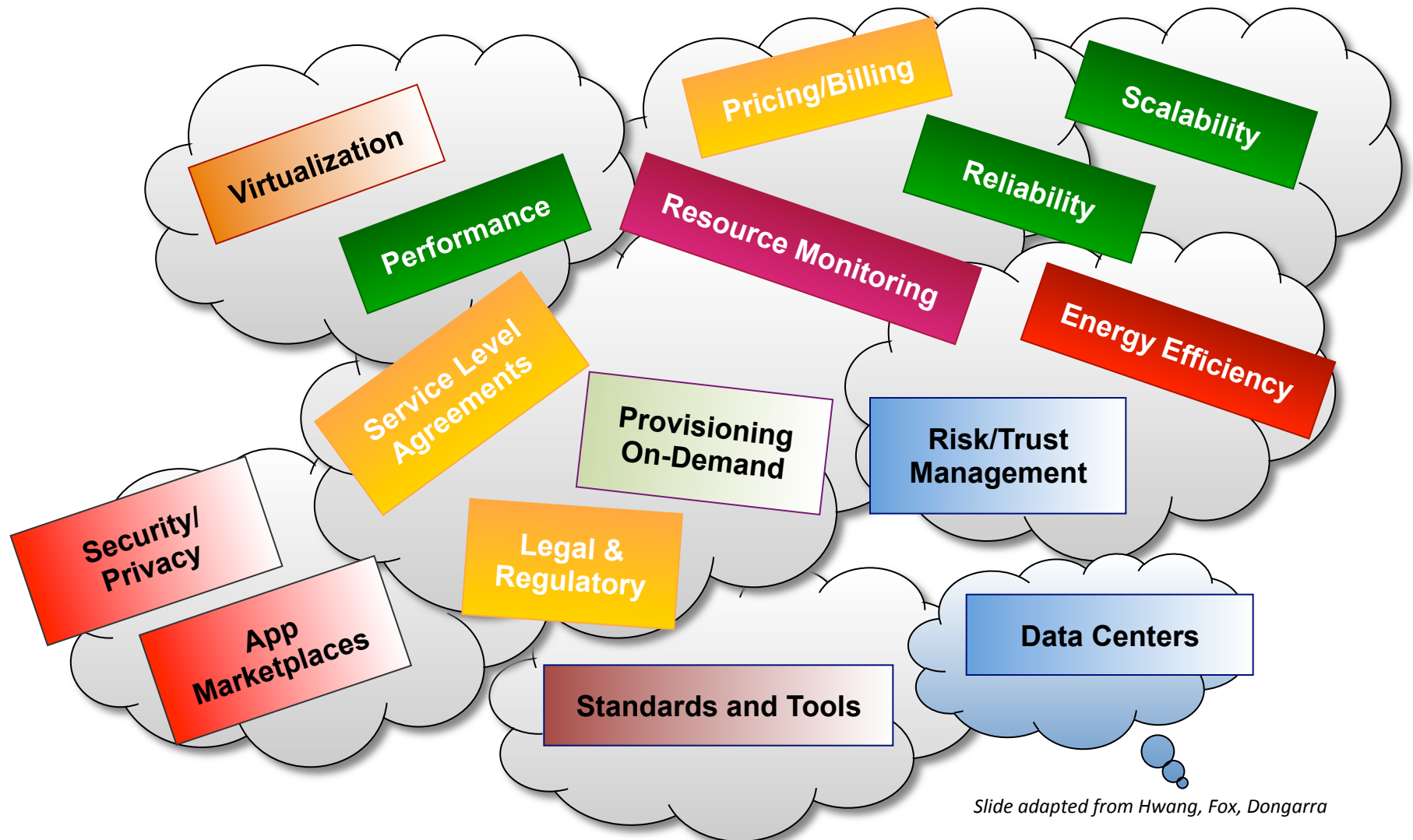
***Assistant Professor, Department of Computer Science***

Talk at GENI Engineering Conference (GEC22),  
Washington DC, March 2015

# What is Cloud Computing?



# Topics covered on cloud computing and networking infrastructure



*Slide adapted from Hwang, Fox, Dongarra*

# Cloud Computing Course Information

*Two classes taught: Fall 2013 and Spring 2015*

- **Pre-requisites:**
  - Object Oriented Programming
  - (Operating Systems-I), (Computer Networks-I), (Cyber Security-I)
- **Students:**
  - **Fall 2013: 23 Graduate, 7 Undergraduate, 5 visitors**
  - **Spring 2015: 24 Graduate, 36 Undergraduate, 5 visitors**
- **Textbook:**
  - Distributed and Cloud Computing - by K. Hwang, J. Dongarra, and G. Fox; First Edition [ISBN: 9780123858801] ([required](#))
- **Teaching Assistants:**
  - Sripriya Seetharam & Ronny Bazan Antequera – attended GENI Summer Camp, GECs, Thesis Research that uses GENI infrastructure
  - Work closely with GPO (Thanks! Niky, Sarah and Vic)
  - Monitors and answers student issues in Blackboard Discussion Forum
- **Uni. of Missouri GENI Rack Tour**

# Cloud Computing Course Objectives

- Be competent with principles and architectures that integrate computing theories and information technologies
- Be competent with distributed system models, computer virtualization concepts, network virtualization concepts
- Be competent with the design, programming and application of *distributed* and *cloud computing* systems

and...

- Be familiar with cyber-security and service-oriented concepts relating to use/design of cloud computing systems
- Be familiar with App customization of cloud computing infrastructures, and standards through hands-on experience

# Course Lab Assignments

- 6 labs to develop technical background and skills for working with distributed system and related software environments
- 3 Amazon Web Services (AWS) labs - <http://aws.amazon.com>
  - Received \$100 usage credit per student through their academic program
  - Focus areas:
    - Instance setup with Cloud Watch alarm for billing alerts
    - Web App setup with EC2, S3 and Cloud Watch
    - Autoscaling to handle user load bursts in a Web App
- **3 GENI labs** - <http://groups.geni.net/geni/wiki>
  - “Calyam UMissouri Fall2013” Project in GENI portal (<http://portal.geni.net>)
  - **Focus areas:**
    - **Slice setup for “Hello GENI” experiment** (available on GENI Wiki)
    - **Instrumentation & Measurement Web App setup with GENI Rack VMs and OnTimeMeasure software** (New but available on GENI Wiki)
    - **OpenFlow controller programming for 2 traffic engineering applications: QoS Control based on enterprise policy, and Load Balancing** (New)

# GENI Lab-1 Q&A

## *Familiarizing students with terminology and concepts*

1. Provide a screenshot of the reserved slice resources in your GENI portal account
2. What would be the added capabilities and benefits in performing an experiment on GENI Future Internet infrastructure versus the commercial Internet?
3. Define: (i) slice, (ii) sliver, (iii) aggregate manager and (iv) Rspec
4. How does Federated Identity and Access Management in the GENI portal allow you to use your Pawprint and password to login? What are the benefits of this approach?
5. Explain the role of “Experiment Control Tools” such as Omni, Gush and Flack in GENI. What are the advantages of using the Omni command line tool in comparison to using the Flack GUI tool?
6. What is the significance of generating the SSH and SSL keys in your .ssh folder?

# Final Course Project

- **Objective:**
  - To develop programming and performance evaluation skills
  - To reinforce the understanding of major concepts in the course
    - *Graduate Student team projects* require use of either C/C++ or Java, as well as cloud-platform specific tools and applications
    - *Undergraduate student participation is optional (extra credit)*
- **Approach:**
  - Solve a “real” problem in a GENI/AWS testbed
    - **Teams:** Core Logic Team, Testbed Team, User Interface Team
    - Must use web services, show design tradeoffs (e.g., *how VM obfuscation for ‘security’ complicates ‘manageability’ of infrastructure*)
    - Build upon Cloud Computing material covered in class
    - Student domain expert for each project
  - In-Class “Circle Time” to provide on-going guidance
  - Final Project Presentations/Demos
  - 25% Grade based on overall team performance, peer-reviews



# Fall 2013 Final Course Projects

- 1. Platform-as-a-Service for City-supported Hybrid Cloud**
  - Based on requirements from “ElderCare-as-a-Service” testbed in GENI
  - Concepts of Desktop-to-Cloud, Recommendation Engine
- 2. Software-defined Networking for Multi-tenancy**
  - Based on requirements from a MU-OSU “Science DMZ” testbed
  - Concepts of Network Virtualization, RoCE 10 Gbps Data Transfer Performance
- 3. Data Integrity Security Protection and Monitoring**
  - Based on requirements from “Thomson Reuters” testbed in GENI/AWS
  - Concepts of Distributed Trust, Anomaly Detection for Just-in-time News Feeds
- 4. App Marketplace for Advanced Manufacturing**
  - Based on requirements from “Simulation-as-a-Service” testbed in GENI/AWS
  - Concepts of App Workflow, Mapping User Requirements to Cloud Infrastructure

# Spring 2015 Final Course Projects

- 1. Custom Image Templates enabled Platform-as-a-Service**
  - Based on requirements from SoyKB testbed in GENI and AWS
  - Concepts of Custom Image Templates, Service Optimization, Workflow Monitoring
  - Domain Experts: *Ronny Bazan Antequera, Yuanxun Zhang*
- 2. Software-defined Networking for Intelligent Data Movement**
  - Based on requirements from GENI Intelligent Data Movement Service
  - Concepts of Network Virtualization, Neighborhood Method
  - Domain Expert: *Dmitrii Chemodanov*
- 3. Moving Target Defense for Resilient Cloud Management**
  - Based on requirements from cyber attack defense testbed in DeterLab
  - Concepts of Cyber Attack Generation, Proactive and Reactive Migration Triggers
  - Domain Experts: *Saptarshi Debroy and Ravi Akella*
- 4. Hosting and Pricing of an Advanced Manufacturing App Marketplace**
  - Based on requirements from “Factory of the Future” testbed in GENI and AWS
  - Concepts of App Workflow Chaining, Resource Brokering, App Pricing Models
  - Domain Expert: *Amit Kumar Rama Akula*

# Conclusion: How GENI helped!

- GENI provided an excellent platform to develop and administer hands-on laboratory exercises for students in the Course on cloud computing topics such as:
  - distributed system resource discovery and management
  - federated identity and access management
  - application/system/network performance measurement and adaptation
  - software-defined networking with OpenFlow
  - experimental testbed setup, related web services, and programming practices
- Student Learning enabled through GENI
  - Perception change of App and their high scalability in Cloud
  - Understanding Architecture diagrams!
  - Hands-on Experience with advanced computing technologies
  - Supplement class discussions of concepts through Lab exercises

# Conclusion: How GENI helped! (2)

## Some Student Quotes

“I am able to visualize the architecture diagrams”

“Labs in GENI were exciting!”

“There are a lot of technologies and terminology...”

“It takes time to understand detailed instructions in labs”

“I no longer view Apps as simply code executing on a single machine”

“I appreciate the fact that there is a ‘user view’ and ‘provider view’...”

“When to use what virtualization is challenging”

## Other thoughts...

- Detailed instructions were needed to complete the labs
  - Using distributed system/network resources and working with tools such as those in GENI/AWS was a new experience
- AWS and GENI combination benefits
  - Many similarities in tools, best practices – reinforced learning!
- GENI Project Office co-ordination was important
  - To determine timing of labs, ensuring resource availability
- Offering a two parts (Cloud Computing I & II) in consecutive semesters to have more in-depth GENI experiments

Thank you for your attention!

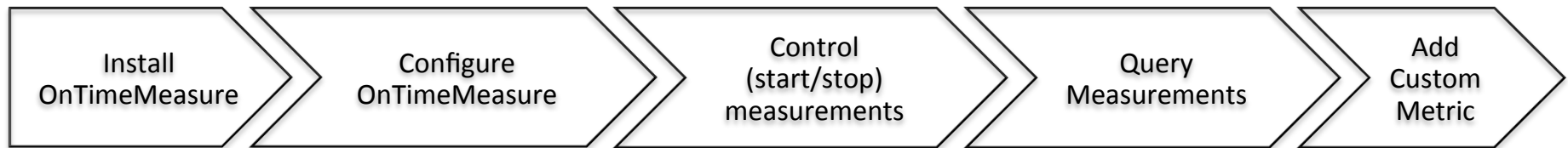


# GENI Lab-1 Exercise



- **Lab # 1 – GENI Account Setup and Services Overview**
- **Purpose of the Lab**
  - Setup your GENI account, and use it to reserve an experiment environment a.k.a. “slice” that features 2 compute nodes communicating over a Layer 2 connection
  - Run a “Hello GENI” experiment with GUI (Flack) and Command-line (Omni) tools in your reserved slice by automatic installation and execution of software to perform basic bandwidth measurement tests between the nodes

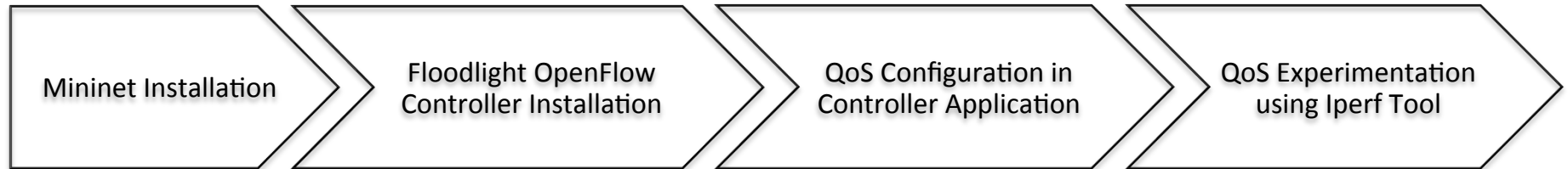
# GENI Lab-2 Exercise



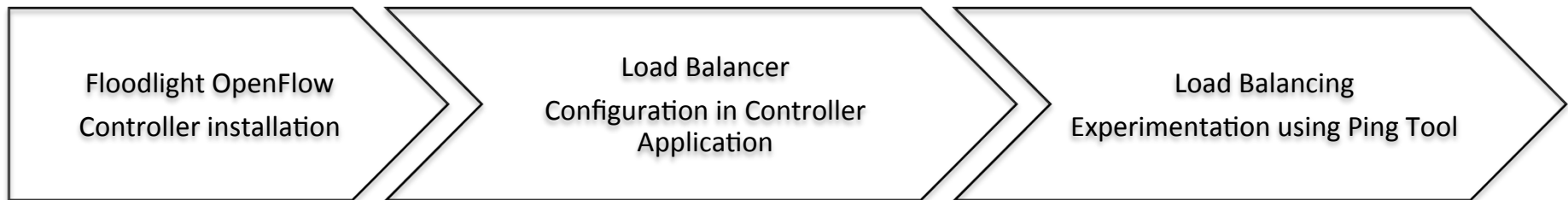
- **Lab # 2 – Instrumentation and Measurement of GENI Slice**
- **Purpose of the Lab**
  - Install and configure a distributed measurement Web App viz., “OnTimeMeasure” within a slice
  - Configuration involves ‘Node Beacons’ at measurement points, and a ‘Root Beacon’ that acts as a central intelligence module
  - Schedule measurements from the Root Beacon and query/visualize performance between two compute nodes that host Node Beacons
  - Performance data collection comprises of network health metrics such as round trip delay, loss, jitter and throughput



# GENI/SDN Lab-3 Exercise



## Lab Experiment #1 (QoS Control through Network-Edge Rate Limiting) Steps Overview



## Lab Experiment #2 (Load Balancer for Scalable Handling of Traffic Flows) Steps Overview

- **Lab # 3 – QoS Configuration and Load Balancing using Software Defined Networking/OpenFlow**
- **Purpose of the Lab**
  - Install and configure Mininet SDN emulator with 2 traffic engineering experiment applications to understand how to program ‘flow spaces’ within networks to: (i) comply with enterprise network capacity provisioning policies, and (ii) balance the utilization of network resources
  - Use Iperf and Ping Tools to verify your SDN functionality