

This two day workshop introduces a Cloud Reference Model and explores various aspects of Cloud solution development. Special attention is given to cloud programming standards and development best practices. Upon completion of this course, students will have an understanding of the Cloud Computing environment and practical experience in designing, developing, and deploying Cloud-based solutions. This hands-on workshop consists of 50% hands-on labs in an actual Cloud Computing development environment, and 50% lecture and discussion. Students should have some hands-on experience developing software using any object-oriented programming language. All labs are developed using the Java Programming Language and supporting Application Programming Interfaces (API).

Course Objectives:

- **Cloud Reference Model**
- **Cloud Layering**
- **Understanding Cloud Development Life Cycle**
- **Programming Standards for the Cloud**
- **Cloud User Interfaces**
- **Service Interface Development**
- **Testing and Building Security in the Cloud**

Audience: Information Technology Developers and Architects

Prerequisites: Foundational Knowledge in Cloud Computing.

Number of Days: 2 days

<p>1</p> <p>Cloud Reference Model</p> <p>Parsing the Cloud Service Model</p> <p>Cloud Reference Model</p> <p>Cloud Infrastructure</p> <p>Cloud Infrastructure – Vendor Comparison</p> <p>Cloud Infrastructure - Cloud Storage</p> <p>Cloud Platform</p> <p>Cloud Software</p> <p>SaaS - Cloud Services</p> <p>SaaS - Cloud Applications</p> <p>OpenStack Solution Stack</p> <p>OpenStack main components/services</p> <p>Compute (Nova)</p> <p>Main Compute (Nova) modules/services</p> <p>Image (Glance)</p> <p>Object Store (Swift)</p> <p>Components of Swift</p> <p>Block Storage (Cinder)</p>	<p>2</p> <p>3</p>	<p>Identity (Keystone)</p> <p>Network (Quantum)</p> <p>Dashboard (Horizon)</p> <p>Pulling It All Together</p> <p>Cloud Layering</p> <p>Cloud Layering</p> <p>Content Services</p> <p>Logic Services</p> <p>Orchestration in the Cloud</p> <p>Utility - Security Services</p> <p>Utility - Data Services</p> <p>Layering Examples</p> <p>SDLC In The Cloud</p> <p>Software Development Lifecycle in the Cloud</p> <p>Requirements Discovery</p> <p>Cloud-Based Analysis and Design</p> <p>Development</p> <p>Implementation and Testing</p>
--	---------------------------------	--

	Monitoring		Additional UI Considerations
	Retirement		Real-time Considerations
4	Cloud Standards		Social Network Considerations
	What Exactly Are We Standardizing?		Cloud Client and Web User UIs
	Standardizing on a Definition		Data Transportation Considerations
	Simple Concept, Difficult		Emerging Standard - WebSocket
	Implementation		Identity Management
	Perspective #1 – Turf Wars		Mashups in the Cloud
	Turf Wars		Handling Error Messages in the Cloud
	Other Groups Defining Standards	7	Programming Cloud Services
	Recent Progress		Conceptual View of Cloud
	Perspective #2 – Let the Market Decide		Access to Cloud Services Remotely
	Standardization is Restrictive not		Conceptual Model of Multi-tiered
	Creative		Services
	Fostering Creativity		Service Elements
	Survival of the Fittest		The Role of Middleware
	Perspective #3 – The Simple View		Thin Client vs Fat Client in The Cloud
	Standards to Date		Thin Client vs Fat Client (cont)
	OCCI Details		Programming for the Cloud: Amazon
	OCCI Client Handshake		Cloud Overview
	OCCI Server Response...		AWS Architecture
	OCCI Cloud Infrastructure Categories		Working with Amazon Web Services
	Best Practices for Working with Cloud		(AWS)
	Standards: Building Cloud		Building an Amazon Cloud Service
	Solutions Today		Microsoft Azure Overview
5	Cloud Development		Azure Architecture
	Implementing Cloud Services		Working with Windows Azure
	Common Pitfalls for Cloud Developers		Building an Azure Cloud Service
	Building Composite Solutions		Google Cloud Overview
	Cloud Development Stacks		Google App Engine Architecture
	Creating Services for Amazon WS		Google Cloud Storage
	Testing in the Amazon Cloud		Working with the Google App Engine
	Deploying Amazon Web Services		Building a Google App
	Consuming Amazon Web Services		Learn By Doing
	Creating Services for OpenStack		Comparing Offerings
	Creating Applications for OpenStack	8	Testing In the Cloud
	Testing OpenStack Solutions		Layered Testing of Cloud Solutions
	Consuming OpenStack Solutions		Phased Testing of Cloud Solutions
	Creating Services for Google		Unit Testing of Cloud Solutions
	Testing Google Cloud Services		Integration Testing of Cloud Solutions
	Deploying Google Services		Exception-based Testing of Cloud
	Consuming Google Services		Solutions
6	Programming Cloud UIS		Boxy Testing...
	Main User Interface Types		Black Box Testing
	Other User Interface Types		Dynamic Black Box Testing
	Primary UI Considerations		Black Box Pros and Cons

White Box Testing
Static Testing
White Box Pros and Cons
Gray Box Testing
Other Types of Testing
9 Securing Cloud Services
Notable Cloud Breaches in 2011
Notable Breach #2
Notable Breach #3
Notable Breach #4 & #5
Notable Breach #6
Top 8 Cloud Security Risks: Top Cloud
 Computing Threats
Abuse/Nefarious Use of Cloud
Insecure Interfaces
Malicious Insiders
Shared Technology Vulnerabilities
Data Loss/Leakage
Data Loss/Leakage for T-Mobile
 Sidekick
Account, Service & Traffic Hijacking
Hijacking Amazon EC2
Unknown Risk Profile
Unknown Risk Profile – Heartland Data
 Breach
8th Risk
Mitigating Cloud Security Risks
Five Mitigation Strategies
Federated ID
Always-on Connectivity
Multi-layer Inspection
Centralized Management
Virtual Desktop Protection
Look toward standards
Security in the Amazon Cloud: Amazon
 Web Services (AWS) Security
 Concerns
AWS Service Specific Security
AWS Cloud ID Service
Working with the AWS Cloud ID
 Service
Protect AWS Data In Transit
Protecting AWS Data at Rest
Security in the Google Cloud
Design with Security in Mind
GAE Cloud Security Module

OpenStack Cloud Perimeter Security
OpenStack Security Groups
Nova Client Security Group Commands
Nova Security Command Examples
Identity Management with Keystone
Keystone command-line
Keystone REST API
Learn By Doing