

# Composition of Semantic Web Service on Cloud : A QoS View

Ratnesh Kushwaha

(M.E.) Computer Engg. SGSITS Indore

*Guided By Prof. L. Purohit (I.T. Department)*

June 26, 2013

# Table of contents

- 1 Problem Domain
- 2 Introduction
- 3 Problem with Simple Web Service
- 4 Proposed Approach
- 5 System Architecture
- 6 Results
- 7 Screenshots
- 8 Conclusions
- 9 Future Work
- 10 References

# Problem Domain

- Different service provider provide different service through cloud.
- Selection of service are done on basis of cost cost.
- Ignore the other QoS of services.
- No proper definition and calculation QoS of cloud is available.

# Introduction

- Purpose of research
  - Virtualization support to cloud service.
  - Define and Calculate QoS of Cloud Service.
  - Customized selection and Composition of Cloud Service.
- Searching service on cloud using Cloud Ontology Description.
- Details of service using semantic description for full potential semantic web service.

# Cloud Definition

Cloud computing is

- ubiquitous
- convenient
- on-demand network

a model for enabling **access to a shared pool of configurable computing resources**

# Three Service Model

- IaaS(Infrastructure as a Service)- basic computing and storage resources Ex. Amazon EC2, vCloud
- SaaS(Software as a Service)- cloud applications Ex. Office 365, Gmail
- PaaS(Platform as a Service)- cloud application infrastructure Ex. Salesforce.com, Windows Azure

Available	Standalone Servers	IaaS	PaaS	SaaS
Applications	No	No	No	Yes
Runtimes	No	No	Yes	Yes
Database	No	No	Yes	Yes
Operating System	No	No	Yes	Yes
Virtualization	No	Yes	Yes	Yes
Server	No	Yes	Yes	Yes
Storage	No	Yes	Yes	Yes
Networking	No	Yes	Yes	Yes

Figure: Comparison

**This project is IaaS type model.**

# Life Cycle of Cloud Service

Service Requirement	Functional Requirement Technical Requirement Budgetary Requirement
Service Discovery	Cloud Service matches with requirement
Service Negotiation	Messages exchanged to establish SLA
Service Composition	Combination of set of services (single virtualized service)
Service Consumption	Delivery of service

Figure: Cloud Life Cycle

# Problem with Simple Web Service

- It specifies appearance, not meaning.
- It is fine if interaction is with human, but if you want your agents to be able to process the information, they need to be able to understand what is on a web page.
- UDDI does not provide complete description.



# Semantic Web Composition

XML	Customized tags
RDF	Relation in triples
Ontologies	Heirarchies of concepts
Inference Rules	Rules for infer

Figure: Semantic Web Composition

- Ontology- A set of statements (and usually rules for reasoning) about the world
- E.g.: Publications (e.g., books, magazines, articles) and their properties (e.g., titles, authors, reviews, reprintings)

# Approach Details

- 1 Develop a cloud.
- 2 Creation of semantic web service.
- 3 Deployed it to virtual servers.
- 4 Searching and Composition of service.
- 5 Calculate cloud QoS and comparing results.

# System Architecture

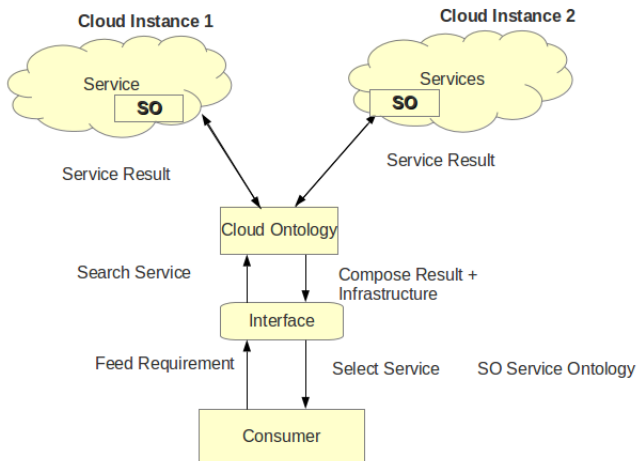


Figure: System Architecture

# QoS Definition and Comparison

In this method, three level of QoS of cloud services are calculated they are

- ① On cloud software components on which cloud is developed.
- ② Infrastructure of instance on which service is deployed.
- ③ Service level response time.

# Cloud Level QoS (Level1)

```
[root@cloud ~]# euca-describe-groups
GROUP 742113722717 default default group
PERMISSION 742113722717 default ALLOWS tcp 22 22 FROM
GROUP 742113722717 MySecurityGroup Ping,http,ssh
PERMISSION 742113722717 MySecurityGroup ALLOWS tcp 8080 8080
PERMISSION 742113722717 MySecurityGroup ALLOWS icmp -1 -1
PERMISSION 742113722717 MySecurityGroup ALLOWS tcp 22 22
[root@cloud ~]#
```

Figure: Security Groups

## Instance QoS(Level 2)

The infrastructure used by instances describe its QoS which includes the CPU, Disk space and number of CPU.

**Table:** Service with their Instance

S.No.	Instance	PublicIP	RAM	DiskSpace	CPU
1	Instance1	10.1.175.110	512	5GB	1
2	Instance2	10.1.175.111	512	5GB	1
3	Instance3	10.1.175.112	1024	10GB	2

## Service Response Time (Level3)

Response Time = Time of receiving Response – Time of making the request

Table: Service with their Instance

S.No.	ServiceName	InstanceUse
1	HotelA	Instance1
2	HotelB	Instance2
3	TrainA	Instance1
4	TrainB	Instance2
5	BusA	Instance1
6	BusB	Instance2
7	FlightA	Instance1
8	FLightB	Instance2

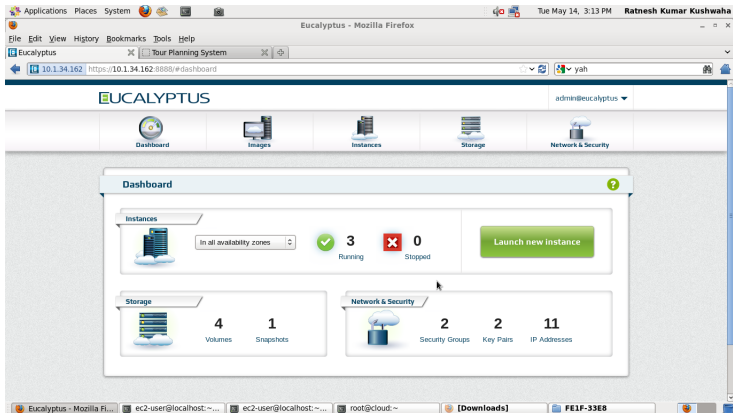
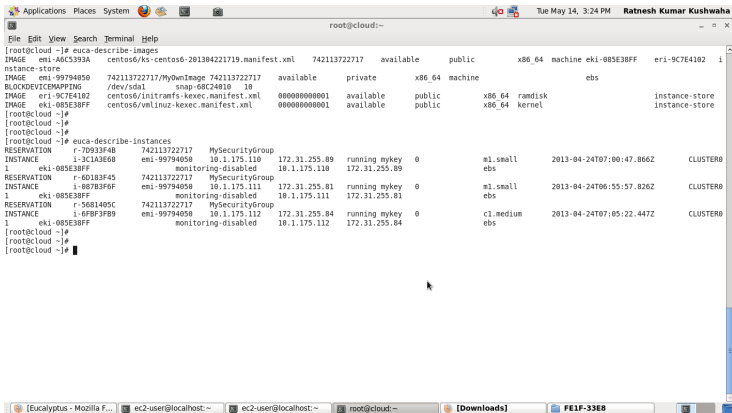


Figure: CloudHomePage



# CloudInstances and Images



```
Applications Places System root@cloudi:~ Tue May 14, 3:24 PM Ratnesh Kumar Kushwaha
File Edit View Search Terminal Help
[root@cloud ~]# euca-describe-images
IMAGE emi-A6C5393A centos6/ks-centos6-201304221719.manifest.xml 742113722717 available public x86_64 machine emi-085E38FF eri-9C7E4102 i
instance-store
IMAGE emi-99794050 742113722717/MyOwnImage 742113722717 available private x86_64 machine ebs
BLOCKDEVICEMAPPING /dev/sda1 snap-68C24810 10
IMAGE eri-9C7E4102 centos6/initramfs-kexec.manifest.xml 000000000001 available public x86_64 ramdisk instance-store
IMAGE eki-085E38FF centos6/vmlinuz-kexec.manifest.xml 000000000001 available public x86_64 kernel instance-store
[root@cloud ~]#
[root@cloud ~]#
[root@cloud ~]# euca-describe-instances
[RESERVATION r-7D933F4B 742113722717 MySecurityGroup
INSTANCE i-3C1A3E68 emi-99794050 10.1.175.110 172.31.255.89 running mykey 0 m1.small 2013-04-24T07:00:47.866Z CLUSTERB
1 eki-085E38FF monitoring-disabled 10.1.175.110 172.31.255.89
RESERVATION r-6D183F45 742113722717 MySecurityGroup
INSTANCE i-867B3F0F emi-99794050 10.1.175.111 172.31.255.81 running mykey 0 m1.small 2013-04-24T06:55:57.826Z CLUSTERB
1 eki-085E38FF monitoring-disabled 10.1.175.111 172.31.255.81
RESERVATION r-5681405C 742113722717 MySecurityGroup
INSTANCE i-6FBF3FB9 emi-99794050 10.1.175.112 172.31.255.84 running mykey 0 c1.medium 2013-04-24T07:05:22.447Z CLUSTERB
1 eki-085E38FF monitoring-disabled 10.1.175.112 172.31.255.84]
[root@cloud ~]#
[root@cloud ~]#
[root@cloud ~]#
```

Figure: Eucalyptus Commands

# Instances Comparison

The screenshot displays a desktop environment with three terminal windows open, comparing system resources across different instances. The desktop background is dark blue, and the taskbar at the bottom shows application icons and a system tray.

**Terminal 1 (Top Left):** `ec2-user@localhost:~`

```
File Edit View Search Terminal Help
[ec2-user@localhost ~]$ free -m
Mem:           497      486      18      0      buffers 1      cached 24
      total    used      free shared      buffers cached
-/+ buffers/cache:
Swap:           0           0           0
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
```

**Terminal 2 (Top Right):** `ec2-user@localhost:~`

```
File Edit View Search Terminal Help
[ec2-user@localhost ~]$ free -m
Mem:           497      488      9      0      buffers 14      cached 123
      total    used      free shared      buffers cached
-/+ buffers/cache:
Swap:           0           0           0
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
[ec2-user@localhost ~]$
```

**Terminal 3 (Bottom):** `[Ratnesh@cloud ~]$`

```
File Edit View Search Terminal Help
[Ratnesh@cloud ~]$ free -m
Mem:           3739      3504      235      0      buffers 562      cach 14
      total    used      free shared      buffers cach
-/+ buffers/cache:
Swap:           3871           0      3871
[Ratnesh@cloud ~]$
[Ratnesh@cloud ~]$
[Ratnesh@cloud ~]$
[Ratnesh@cloud ~]$
[Ratnesh@cloud ~]$
[Ratnesh@cloud ~]$
[Ratnesh@cloud ~]$
[Ratnesh@cloud ~]$
[Ratnesh@cloud ~]$
[Ratnesh@cloud ~]$
```

Figure: Eucalyptus Commands

# HomePage

The screenshot shows a web browser window titled "Tour Planning System - Mozilla Firefox". The address bar shows the URL "10.1.175.110:8080/MyTourism/Exp.jsp". The page has a yellow header with the text "Tour Planning System on Cloud" and "SGSITS Indore". Below the header, there is a navigation bar with buttons for "Trains", "Flights", "Roadways", "Hotel Accomodation", and "QoS Comparison". The "Hotel Accomodation" button is selected. The main content area contains a form with the following fields and buttons:

- From: CityA (dropdown menu)
- To: CityB (dropdown menu)
- Date: 2010-10-10
- Number of travelers: 1
- Search Hotels (button)
- Number of Rooms: 0
- Search Buses (button)

The browser's taskbar at the bottom shows several open windows, including "Tour Planning System...", "ec2-user@localhost:~...", "root@cloud:~...", "[Downloads]", and "FE1F-33E8".

Figure: HomePage

# SearchResult

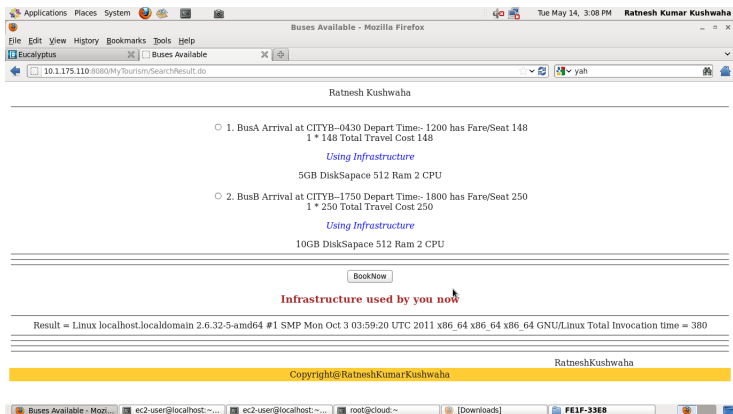


Figure: SearchResult

# ResponseTimeGraph1

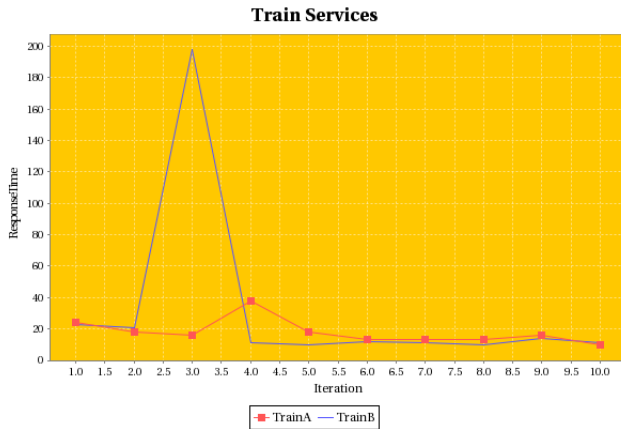


Figure: Graph1

# ResponseTimeGraph2

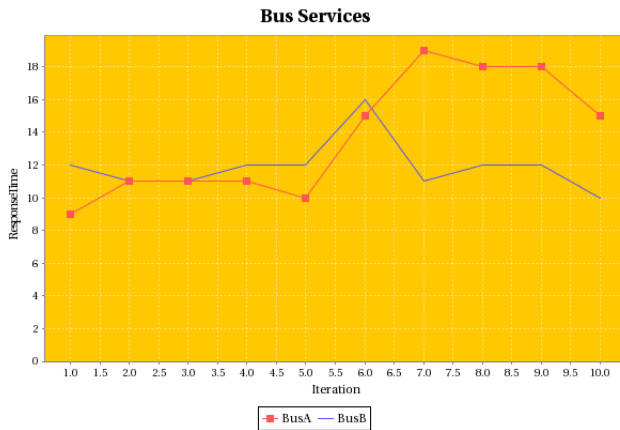


Figure: Graph2

# Conclusion and Limitation

- The experimental results show that ontological description is best for searching cloud services.
- To describe QoS we need three level of comparison.

## **Limitations**

- Limitation of ontology (Updating Problem).
- Limitation OWL API.

## Future Work

The present work uses ontologies to describe web service. The agents uses this description for composition of web service within a cloud using cloud instance. The work can be enhanced by composing service on different clouds. These clouds should be developed using different technologies for better comparison and results.



# References

- Introduction to WSDL,w3schools, Available at:[http://www.w3schools.com/wsdl/wsdl\\_intro.asp](http://www.w3schools.com/wsdl/wsdl_intro.asp).
- Cloud Computing Nariman (nmirzaei@indiana.edu) Fall 2008
- Apache Axis Website  
<http://axis.apache.org/axis2/java/core/>
- Apache JUDDI Website <http://ws.apache.org/juddi/>
- Protege Website <http://protege.stanford.edu/>
- Eucalyptus Open Source,Eucalyptus,<http://www.eucalyptus.com/eucalyptus-cloud>
- "SOAP Introduction,w3schools,Available at [http://www.w3schools.com/soap/soap\\_intro.asp](http://www.w3schools.com/soap/soap_intro.asp).

# References

- Wang Qing-Ming, Tang Yong, Zhang Zan-Bo, Research in Enterprise Applications of Dynamic Web Service Composition Methods And Models, Preceding of Second International Symposium on Electronic Commerce and Security, IEEE 2009, pp 146-150.
- Kwang Mong Sim. Agent based cloud computing. IEEE TRANSACTIONS ON SERVICES COMPUTING, 2010.
- Sandeep Kumar and Nikos E Mastorakis. Novel models for multi-agent negotiation based semantic webservice composition. WSEAS TRANSACTIONS on COMPUTERS,, 2010.
- Peter Mell, Timothy Grance, NIST Definition of CCloud, NIST SpecialPublication 800-145 2011.

# Thank You

Presentation Prepared Using  $\text{\LaTeX}$