Performance Evaluation of Round Robin Algorithm in Cloud Environment

Asha M L
Assistant Professor, Dept. of CSE
SVCE, Bangalore

Neethu Myshri R
M.E (CSE), Dept. of CSE
University Visvesvaraya College of Engineering,Bangalore

ABSTRACT
Cloud computing is an emerging technology and trend in research work. In this paper we are using a novel methodology of using a cloud analyst toolkit to simulate and understand the behavior of cloud computing and deployment models. Cloud gives the services through the internet managing the resources is complex job, load balancing is very much essential to perform the task. Here we are using cloud Analyst simulator. To understand the complexity of load by evaluating the algorithm called Round robin Algorithm. It gives an idea for the cloud users to choose which data center is better in cost on large area internet.

***Keywords*:** Data center, Load Balancing policy, Response time.

1. INTRODUCTION
Cloud computing is the use of computing resources that are delivered as a service over a network. Cloud computing is essentially internet-based computing which allows users access to a number of virtual services and resources that can be utilized for exchange between devices on demand. The 'internet cloud' as it is known therefore gets rid of the need for any storage hardware thereby allowing for a business's it infrastructure and devices to be smaller, faster, and more energy efficient.

The Virtualization gives foundation for cloud technology,[5] It is like a computer programs through an internet connection rather than installing application on every office computer. Virtualization helps to access the data or server without knowing any details. Virtualization gives the services to user such as infrastructure, storage, software as services.

The virtualization forms the foundation of cloud technology where Virtualization is an emerging IT paradigm that separates computing functions and technology implementations from physical hardware. Cloud computing, for example, is the Virtualization of computer programs through an internet connection rather than installing applications on every office computer. Using virtualization, users can access servers or storage without knowing specific server or storage details. The Virtualization layer will execute user request for computing resources by accessing appropriate resources. Virtualization can be applied to many types of computer resources: Infrastructure such as Storage,

Network, Compute (CPU / Memory etc.), Platform (such as Linux/ Windows OS) and Software as Services. Cloud computing in computing research and industry today has the potential to make the new idea of 'computing as a utility in the near future. The Internet is often represented as a cloud and the term "Cloud Computing". Cloud computing is the dynamic provisioning of IT capabilities/IT services (hardware, software. Cloud computing has several characteristics. The major ones are [6]:

1. Dynamic provisioning
2. Network Access
3. Shared infrastructure

The basic service models of the cloud are:

- **Software as a Service(SaaS)-**user wants a service From the cloud, necessary information required to the user is hosted on to the cloud. Customer gets the service from the cloud.
- **Platform as a Service (PaaS) -**user gets access as a platform as a service through the internet. Customer can deploy his own software and applications in the cloud.
- **Infrastructure as a Service (Iaas) -** user gets the benefit of operating system, applications, storage and network connectivity as a service, through cloud infrastructure as a service.

2. ALGORITHM USED
A. Service Broker Policies are

- Closest Data center
- Optimize response time
- Reconfigure dynamically with Load.

B. Load Balancing Techniques

- i)Throttle Load Balancer
- ii) Round robin Algorithm

2.1 Service Proximity based Routing
This is the simplest Service Broker implementation.

1. Service Proximity Service Broker maintains an index table of all Data Centers indexed by their region.
2. When the Internet receives a message from a user base it queries the Service Proximity Service Broker for the destination Data Center Controller.
3. The Service Proximity Service Broker retrieves the region of the sender of the request and queries for the region proximity list for that region from the Internet Characteristics.

This list orders the remaining regions in the order of lowest network latency first when calculated from the given region.

4. The Service Proximity Service Broker picks the first data center located at the earliest/highest region in the proximity list. If more than one data center is located in a region, one is selected randomly

Closest Data center how it takes is:

In this case the proximity is the quickest path to the data center from a user base based on network latency. The service broker will route user traffic to the closest Data center in terms of transmission latency [2].

Load balancing techniques what we are using is.

- Round Robin.
- Equally spreading the current execution.
- Throttled Load Balancer.

This algorithm provides an efficient on demand access to the resources based on his priority factor. When the new request arrives from client, the cloud manager keeps track of each request and available resources and if available, it will allocate them. If the requested resources are not available then it will place the request in queue, which is a collection of all requested clients. When a resource becomes free then the cloud manager will estimate the priority factor and based on that it will allocate the resource to the clients. In this paper we are concentrating on throttled load balancer with two cases.

2.2 Round Robin Algorithm

This load balancing policy [8] attempts to maintain equal workloads on all the available VMs. The algorithm used is quite similar to the throttled case:

1. Active Vm Load Balancer maintains an index table of VMs and the number of requests currently allocated to the VM. At the start all VM's have 0 allocations.

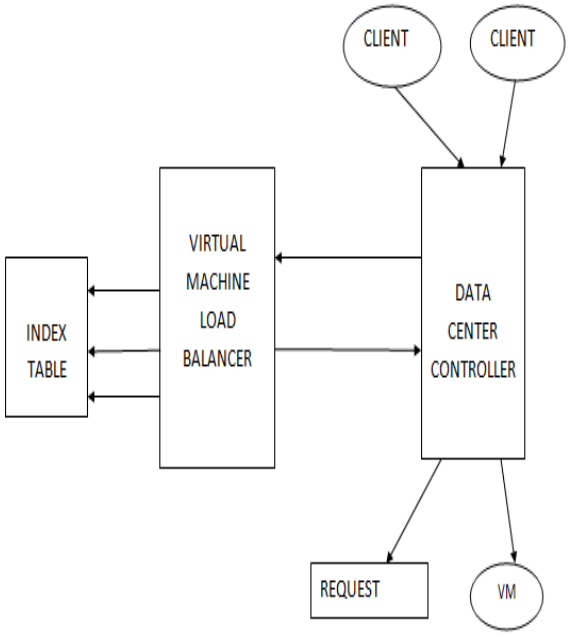

Fig.1. Architecture of Round robin Algorithm

2. When a request to allocate a new VM from the Data Center Controller arrives, it parses the table and identifies the least loaded VM. If there are more than one, the first 19 identified is selected.
3. Active VmLoad Balancer returns the VM id to the Data Center Controller
4. The Data Center Controller sends the request to the VM identified by that id.
5. Data Center Controller notifies the Active VmLoad Balancer of the new allocation
6. Active VmLoad Balancer updates the allocation table increasing the allocations count for that vm
7. When the VM finishes processing the request, and the Data Center Controller receives the response cloud let, it notifies the Active VmLoad Balancer of the VM de-allocation.
8. The Active VmLoad Balancer updates the allocation table by decreasing the allocation count for the VM by one.
9. Continue from step 2

2.3. Calculating the Data Transmission Delay

The data transmission delay is calculated using the following formula:

Ttotal = Tlatency + Ttransfer

Where Tlatency is the network latency and Ttransfer is the time taken to transfer the size of data of a single request (D) from source location to destination. Tlatency is taken from the latency matrix (after applying Poisson distribution on it for distributing it) held in the Internet Characteristics.

Ttransfer = D / Bwperuser

Where

Bwperuser = Bwtotal / Nr

Bwtotal = the total available bandwidth (held in the Internet Characteristics) and
Nr = the number of user requests currently in transmission.
The Internet Characteristics also keeps track of the number of user requests in-flight between two regions for the value of Nr.

3. CLOUDSIM

CloudSim is a simulator toolkit developed by the GRIDS laboratory of University of Melbourne which enables seamless modelling, simulation and experimenting on designing Cloud computing infrastructures. CloudSim is a self-contained platform which can be used to model data centers, service brokers, scheduling and allocation policies of a large scaled Cloud platform. It provides a virtualization engine with extensive features for modelling the creation and life cycle management of virtual engines in a data centre. CloudSim framework is built on top of GridSim framework also developed by the GRIDS laboratory [3]

3.1. Observations of Cloud Analyst

The following observations are made after simulation.

a. Average, minimum and maximum response time in terms of msecs.
b. Average, minimum and maximum Data center processing time in msecs
c. Response time by region.
d. User base hourly average response time.
e. Data center request serving time
f. Data center loading
g. Cost
   - Total VM cost
   - Total data transfer cost
   - Grand total

3.2. Software used

a. Java – The simulator is developed 100% on Java platform, using Java SE 1.6.
b. Java Swing – The GUI component is built using Swing components.
c. CloudSim – CloudSim features for modelling data centers is used in Cloud Analyst.
d. SimJava – SimJava is the underlying simulation framework of CloudSim and some of SimJava are used directly in Cloud Analyst features [3]

3.3. Data Configuration

The configuration options on the main tab are:
User Bases Table – This is a table listing out all the user bases in the simulation. Each user base has following configurable fields, represented by a single row in the table.

a. Name
b. Region
c. Requests per user per hour
d. Data size per request
e. Peak hours
f. Average users during peak hours
g. Average users during off-peak hours

The Add and Remove buttons next to the table can be used to add or remove user bases from the configuration.

4. ANALYSIS

Case 1: we are using many user bases requesting many data centers. It will give the result based on the service broker policy called throttle Load balancer technique.

Case 2: we are using many user bases requesting many data centers. It will give the result based on the service broker policy and algorithm called Round Robin policy.

If many user bases requests for the same database service using service broker policy called Closest data center, then if data center is present in userbase1 and we are requesting services from different regions, then it gives services based on closest Data center. In advance settings number of user bases ten and grouped into single user base then request grouping factor in data center is also grouped as ten, executable instructions per second is 100. Then load balancing techniques we set it as Round Robin policies. Internet characteristic we consider as Transmission delay tween region, units in ms and bandwidth available between regions. Then we analyze total VM cost and total data center cost.
Robin policies and Throttle load balancing policies.
Analysis is done in such a way that,

Overall response time by considering Round Robin is given below:

Ub1: Response time
Avg: 501 ms
Max: 60,1ms
Min 38.9 ms

Ub2: Response time
Avg: 200.6ms
Max: 250ms
Min 158 ms

Ub3: Response time
Avg: 501 ms
Max: 60,1ms
Min 38.9 ms

Ub4: Response time
Avg: 499.6ms
Max: 597ms
Min: 402 ms

Overall response time by considering Throttle Load Balancer given below:

Ub1: Response time
Avg: 485 ms
Max: 60,1ms
Min 38.9 ms

Ub2: Response time
Avg: 300ms
Max: 350ms
Min 190ms

Ub3: Response time
Avg: 505 ms
Max: 70.1ms
Min: 48.9 ms

Ub4: Response time
Avg: 500
Max: 600ms
Min: 490ms

5. EXPERIMENTS AND RESULTS

The results screen will list out the data collected from the simulation. This includes:
1. Overall response time summary (for all the user bases)
2. Response time by user base in tabular format
3. Response time by user base in graphical format broken down into the 24 hours of the day
4. Request servicing time by each data canter in tabular format
5. Request servicing time by data centre in graphical format broken down into 24 hours of the day
6. Data centre loading (number of requests serviced) in graphical format broken down in msecs.
7. Cost details

6. METHODS COMPARED

In this paper we are comparing with two different cases. In first case service Broker policy[10] we are using here as closest data center and Load balancing techniques what we are using here is Throttle Load balancer and comparing average response time with different Load balancing techniques called Round robin technique and comparing average response time for each individual user bases.

7. EVALUATION

Fig 2: Representation main window after simulation Using case 1 with single data center and many user bases

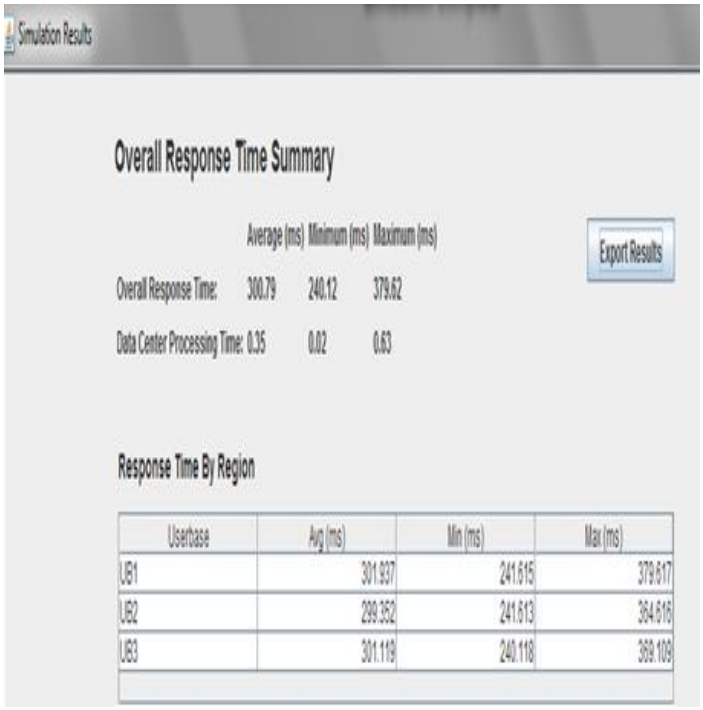

Fig 3: Resultant table for case 1

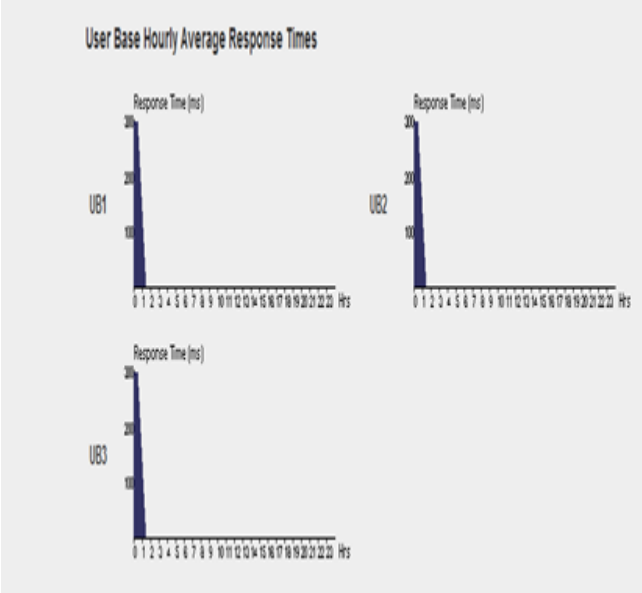

Fig 4: Graph for case 1

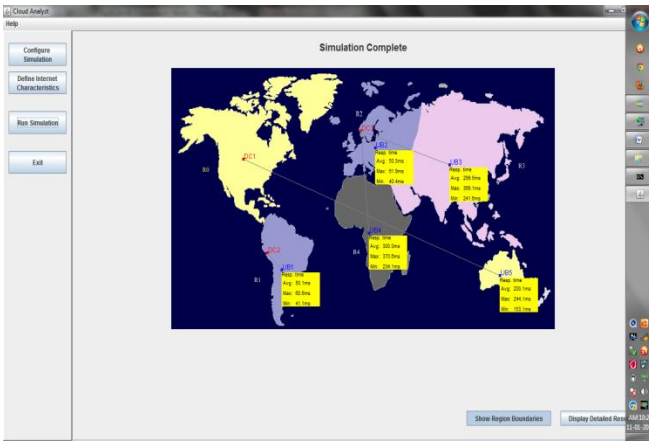

Fig 5: Representation main window after simulation Using case 2 with many data center and many user bases

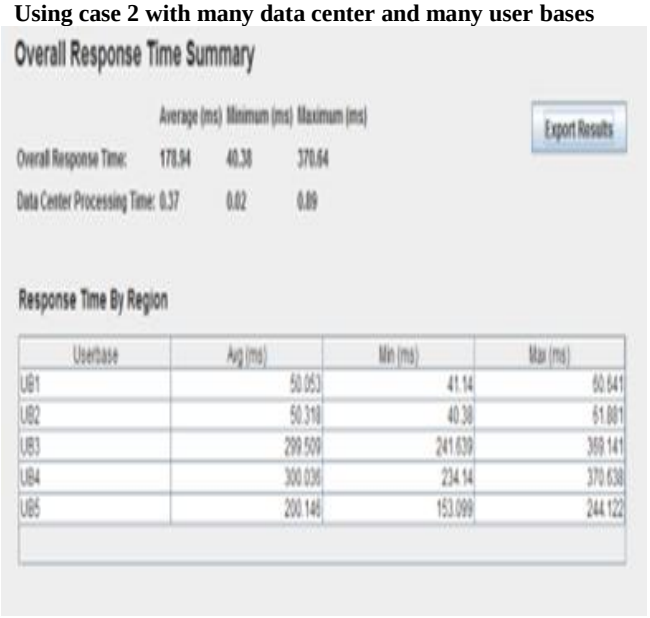

Fig 6: Resultant table and graphs for case 2

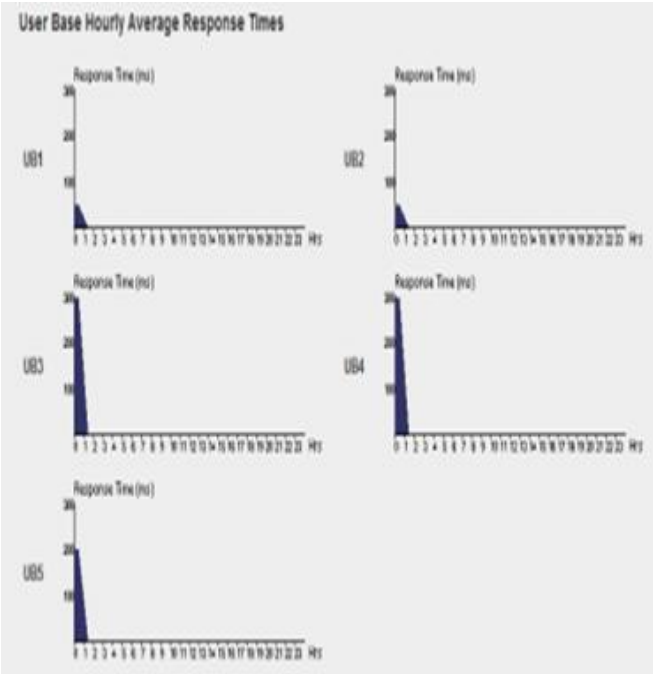

Fig 7: Final result

8. CONCLUSIONS AND FUTURE WORK

In this paper we are concluding the cloud environment with different techniques by service Broker policy by giving different services to user requests; here we are concentrating on closest Data center. If same requests has come from different Data centers from different user Bases, then service will be provided by identifying which is nearest to Data center. And also we are using some advance techniques like how many instructions has to be executed per second , applying particular load balancing techniques like Equally spread current execution load, Throttled Load Ba lancer and here we are concentrating on Round Robin Policy means services of different users will be provided based on Fixed time slices called Round Robin policies. We are also finding different Internet characteristic like Transmission Delay, Bandwidth available for different regions. Overall response time by region and average processing time for Data center loading.

Our future Enhancement is applying green computing for different data centers to save cost and reduce the cost of Load. And future enhancement is how to incorporate the failure mechanism.

9. REFERENCES

[1] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic, Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for delivering Computing as the 5th Utility, Future Generation Computer Systems, Volume 25, Number 6, Pages: 599-616, ISSN: 0167-739X, Elsevier Science, Amsterdam, The Netherlands, and June 2009.

[2] Rajkumar Buyya, Rajiv Ranjan and Rodrigo N. Calheiros, Modelling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities, Proceedings of the 7th High Performance Computing and Simulation Conference (HPCS 2009, ISBN: 978-1-4244-4907-1, IEEE Press, New York, USA), Leipzig, Germany, June 21-24, 2009

[3] Cloudsim based tool for modeling and analysis of large scalecomputing by Bhathiya Wickremasinghe MEDC project, 2009

[4] Wikipedia.org

[5] Efficient load balancing Algorithm for a cloud computing Environment Jasmin James, 38 Sector-A, 2012

[6] Introduction to cloud computing by www.dialogic.com

[7] Platforms for Building and Deploying Applications for Cloud Computing, by Rajkumar Buyya and Karthik Sukumar

[8] Alberto F. De Souza and Rajkumar Buyya, Computer Architecture and High Performance Computing,International Journal of Parallel Programming, Vol. 36, No. 1, Springer, Netherlands, April 2008.

[9] Ivona Brandic and Rajkumar Buyya, Recent Advances in Utility and Cloud Computing, Future Generation Computer Systems, Volume 28, No. 1, ISSN: 0167-739X, Elsevier Press, Amsterdam, The Netherlands, Jan. 2012.

[10] Jörn Altmann, Omer Rana, Rajkumar Buyya, Economics of Computing Services, Future Generation Computer Systems, Volume 28, No. 8, Pages: 1283-1284, ISSN: 0167-739X, Elsevier Press, Amsterdam, The Netherlands, October 2012.