Cost effective selection of Data center by Proximity-Based Routing Policy for Service Brokering in Cloud Environment

BY

Devyaniba Chudasama*
Naimisha Trivedi*
Richa Sinha**

*(Department of Information Technology, Kalol Institute Of Research, Kalol)
***(Asst. Prof., Department of Information Technology, Kalol Institute Of Research, Kalol)

Abstract

Cloud Services may be offered as public or private or combined. There is the demand of timely, repeatable, and controllable methodologies for evaluation of algorithms, applications, and policies before actual development of cloud products. CloudAnalyst is a tool that helps developers to simulate large-scale Cloud applications with the purpose of understanding performance of such applications under various deployment configurations. CloudAnalyst deploys different service brokering policies depending on the requirements of the cloud application. The service proximity based routing policy used in the simulator selects data center from the earliest region to route the user requests. When there is the situation to select one data center from those with same region, the policy selects data-center randomly without considering cost-effectiveness or other parameters. we propose an enhanced proximity-based routing policy that select cost effective data center.

Keywords - Cloud Computing, Simulation, CloudSim, CloudAnalyst, Internet-Cloudlet, Region, Service Broker

1. OVERVIEW

Cloud Computing is getting advanced day by day. Cloud service providers are willing to provide services using large scale cloud environment with cost effectiveness. Also, there are some popular large scaled applications like social-networking and ecommerce.

These applications can benefit to minimize the costs using cloud computing.

Cloud computing is modeled to provide service [1] rather than a product. Services like computation, software, data access and storage are provided to its user without its knowledge about physical location and configuration of the server which is providing the services. Cloud works on the principle of virtualization of resources with on-demand and pay-as-you go model policy. End – user does self-service to access an available pool of computing resources to does its job in just few minutes instead of taking months.

For Cloud Computing environment, simulation based approaches offer significant benefits, as it allows to test cloud services/products in repeatable and controllable environment free of cost, and to tune the performance bottlenecks before deploying on real Clouds. [1]

Next section, we have discussed the general features of simulation tool CloudAnalyst, which is based on the CloudSim.

2. INTRODUCTION TO CLOUDANALYST

CloudAnalyst [1], built on CloudSim, accepts information of geographic location of users generating traffic and location of data centers, and number of resources in each data center as parameters and produces results in the form of XML files so the experiments can be repeated.

Architecture and Component

Components The CloudAnalyst tool has the following components

GUI Package. It is responsible for the graphical user interface, and acts as the front end controller for the application, managing screen transitions and other UI activities.

Simulation. This component is responsible for holding the simulation parameters, creating and executing the simulation.

UserBase. This component models a user base and generates traffic representing the users.[3]

DataCenterController. This component controls the data center activities.

Internet. This component models the Internet and implements the traffic routing behavior.

InternetCharacteristics. This component maintains the characteristics of the Internet during the simulation, including the latencies and available bandwidths between regions, the current traffic levels, and current performance level information for the data centers.

VmLoadBalancer. This component models the load balance policy used by data centers when serving allocation requests. Default load balancing policy uses a round robin algorithm, which allocates all incoming requests to the available virtual machines in round robin fashion without considering the current load on each virtual machine. Additionally, it is possible application of a throttled load balancing policy that limits the number of requests being processed in each virtual machine to a throttling threshold. If requests are received causing this threshold to be exceeded in all...
available virtual machines, then the requests are queued until a virtual machine becomes available. [2]

CloudAppServiceBroker This component models the service brokers that handle traffic routing between user bases and data centers. The default traffic routing policy is routing traffic to the closest data center in terms of network latency from the source user base.

3. PREVIOUS THEORIES
Many related study and work has been proposed for economical Data centers selection in cloud environment.

“Data center is a Computer”[2]. Rule Based Resource Manager proposed for the Hybrid environment, which increase the scalability of private cloud on-demand and reduce the cost. General optimization framework is developed that considers bandwidth usage and electricity costs to solve the datacenter selection problem for cloud services.[3] CloudAnalyst deploys different service brokering policies depending on the requirements, proximity-based routing policy selects nearest data center.[2] Research issues of data center: Flexible Mapping of Services to Resources, Economic Models Driven Optimization Techniques, Cost Model for Mobile Cloud Computing[4]. Service brokering policy that aims to reduce overloading of the closest data center by redirecting a part/whole of the user requests to the next neighboring data center in the same region.[5]

Problems using Service Proximity based Routing
1) Random selection of data center when more than one data center in the same region
2) Possibility of selection of data center with higher cost
3) For the same configuration, results may be different (random selection) and developers/researchers may get difficulties to use the results.

4. SYSTEM MODEL
In Cloud-Analyst, how the routing of user request takes place is shown in the figure (figure 2) below including the use of service broker policy and the virtual machine load balancer. [3]

User Base generates an Internet Cloudlet, with the application id for the application it is intended and also includes the name of the User Base itself as the originator for routing back the responses. With the Zero delay, REQUEST is sent to the Internet. On receiving the REQUEST, Internet consults the Service broker for the data center selection. The service broker uses any one of the service broker policy based on the REQUEST information and sends information about selected data center controller to the Internet. Using this information, Internet sends the REQUEST to the Data Center Controller. Now Selected Data Center Controller uses virtual machines load balancer and after processing the requests, sends the RESPONSE to the Internet.

Now Internet will use the “originator” field of the cloudlet information it received earlier and will add appropriate network delay with RESPONSE and sends to the User Base.

Working of Service Proximity Based Routing
This is the simplest Service Broker implementation. The region selection is based on the earliest/ highest region in the proximity list and any data center of the selected region is then selected randomly for the user requests to be processed. [2]

5. ENHANCED SERVICE BROKER POLICY
we propose an enhanced proximity-based routing policy that avoids the direct selection of nearest data center. If more than one data center is located in a region, the data center having less cost (here, considering only Data transfer cost) will be selected. Now request will be sent to this most cost.

6. SIMULATION CONFIGURATIONS AND RESULTS
Case: 1
Two Data Centers in Same Region
Simulation Duration: 24 Hours
Application Deployment environment
Service Broker Policy: Closest Data Center.

Table 1: Application Development Configuration

<table>
<thead>
<tr>
<th>DATA CENTER</th>
<th>NO OF VM</th>
<th>MEMORY</th>
<th>BANDWIDTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC1</td>
<td>5</td>
<td>512</td>
<td>1000</td>
</tr>
<tr>
<td>DC2</td>
<td>5</td>
<td>512</td>
<td>1000</td>
</tr>
</tbody>
</table>

Table 2: Data Center Configuration

<table>
<thead>
<tr>
<th>NAME</th>
<th>REGION</th>
<th>MEMORY COST $/S</th>
<th>DATA TRANSFER COST $/Gb</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC1</td>
<td>5</td>
<td>0.05</td>
<td>0.1</td>
</tr>
<tr>
<td>DC2</td>
<td>5</td>
<td>0.05</td>
<td>0.2</td>
</tr>
</tbody>
</table>

Table 3: User Bases
User Grouping Factor in User Base: 100
Request Grouping Factor: 100
Executable instruction length per request (bytes): 250

Table 4: Cost

<table>
<thead>
<tr>
<th>DATA CENTER</th>
<th>MEMORY COST $/S</th>
<th>DATA TRANSFER COST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>COST $/Gb</td>
</tr>
<tr>
<td>DC2</td>
<td>21</td>
<td>478.69</td>
</tr>
<tr>
<td>DC1</td>
<td>20</td>
<td>420</td>
</tr>
</tbody>
</table>

Cost
Total Data Transfer Cost ($) : 898.69
Grand Total: ($) 939.69

From the results above, we can conclude that if the data center selection goes randomly, there is no surety about cost effectiveness. We can observe (TABLE 2) that even if the cost of data transfer cost in DC2 is more than that in DC1, overall cost increases. We can also take other parameters under consideration that are not taken in this so called random selection of data center within same region.

Case: 2

Previous case one Data center (DC2) is chosen as per proposed policy. We have assumed that the DC1 is selected as it has less DT cost than DT cost in DC2. Based on this, if DC1 is selected, following is the result for that.

Results: Cost
Total Memory Cost ($) : 41.00
Total Data Transfer Cost ($) : 477
Grand Total: ($) 967.27

Table 4: Cost

<table>
<thead>
<tr>
<th>DATA CENTER</th>
<th>MEMORY COST $/S</th>
<th>DATA TRANSFER COST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>COST $/Gb</td>
</tr>
<tr>
<td>DC1</td>
<td>41</td>
<td>477</td>
</tr>
</tbody>
</table>

From the results above, we can observe that data transfer cost is $ 477.00 for the same requests we used in the case of case:1 and is more cost-effective than just Service Proximity Routing results.

### 7. APPLICATIONS

As the use of cloud computing is being increased day by day, developers and researchers need to test the cloud services/products before its real implementation on the cloud. CloudAnalyst can be used for this purpose.

**Cloud Environment Configuration/Modeling**

Our proposed algorithms are included in the CloudAnalyst which are for cost effective request routing. So, developers and researchers can use CloudAnalyst with those algorithms to configure their cloud environments.

**Consistent Results: No need to repeat the simulation for the same configuration**

Proposed algorithm removes the “random” selection of data centers and so researchers and developers will not get the different results for the same configuration.

### 8. CONCLUSION

We have proposed a new strategy that can be included in the Cloud-Analyst to have cost effective results and development and we can conclude from the results that this strategy is able to do so. From the work done, we can conclude that the simulation process can be improved by modifying or adding new strategies for traffic routing, load balancing etc. to make researchers and developers able to do prediction of real implementation of cloud, easily.

### REFERENCES


[3] Cost Efficient Datacenter Selection for Cloud Services
   Hong Xu, Baochun Li henryxu, bli@eecg.toronto.edu
   Department of Electrical and Computer Engineering
   University of Toronto
