Enhanced Proximity-Based Routing Policy for Service Brokering in Cloud Computing

A. Sarfaraz Ahmed[1]
[1]Department of Computer Science and Engineering
B.S. Abdur Rahman University, India

Abstract – Cloud computing delivers infrastructure, platform, and software (applications) as services, which are made available to consumers as subscription-based services under the pay-as-you-go model. 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 proximity-based routing policy selects nearest data center without taking into account the response time. At times, the proximity-based routing policy may cause overloading of the closest data center. I propose an enhanced proximity-based routing policy that redirects a part/whole of the traffic to the next nearest datacenter in the same region and thereby, reduces overloading of the closest datacenter and avoids SLA violation.

Keywords - Cloud Computing, Simulation, Modeling, CloudSim, CloudAnalyst, Service Broker, Green Computing.

I. Introduction

In 1969, Leonard Kleinrock [2], one of the chief scientists of the original Advanced Research Projects Agency Network (ARPANET) which seeded the Internet, said: "As of now, computer networks are still in their infancy, but as they grow up and become sophisticated, I will probably see the spread of "computer utilities" which, like present electric and telephone utilities, will service individual homes and offices across the country." Cloud Computing has shown tremendous growth both in academia as well as in industry. Cloud Computing offers distributed, virtualized, and elastic resources to the end users and has succeeded in providing the computing resource as a utility.
Study of such distributed, virtualized, and elastic resources can be carried out in a controlled manner with simulation. CloudAnalyst [1] is a tool based on CloudSim [3] which supports visual modeling and simulation of large-scale applications that are deployed on Cloud Architectures.

In section II, I have discussed the general features of CloudAnalyst, which is based on the CloudSim.

In section III, I have discussed the architecture and important components of the CloudAnalyst tool.

In section IV, I have discussed about handling of user request and service brokering policies.

In Section V, I have discussed about the working of proximity-based routing policy.

In Section VI, I have proposed the enhanced proximity-based routing policy.

In Section VII, I have discussed about the simulation configuration and results and finally, I have concluded the results.

II. CloudSim and 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.

III. CloudAnalyst Components

The CloudAnalyst tool has the following components

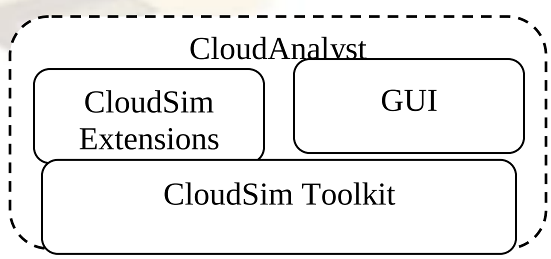

Fig 1: CloudAnalyst architecture

A. Simulation and Results: CloudAnalyst is capable of producing the simulation results in the form of tables and charts. The simulated output will have the response time of the simulated application, overall and average response time of all the user requests simulated. The simulated output will also have the mean time taken by each data center to satisfy each user request; average, minimum, and maximum request processing time by each data center.

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.

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.

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.

IV. Service Broker

Service Broker: Service Broker in CloudAnalyst is responsible for routing the traffic from the user requests from the user base based on the service brokering policies. Service Broker Policies also controls the section of total user base that is being serviced by a data center at a given time.

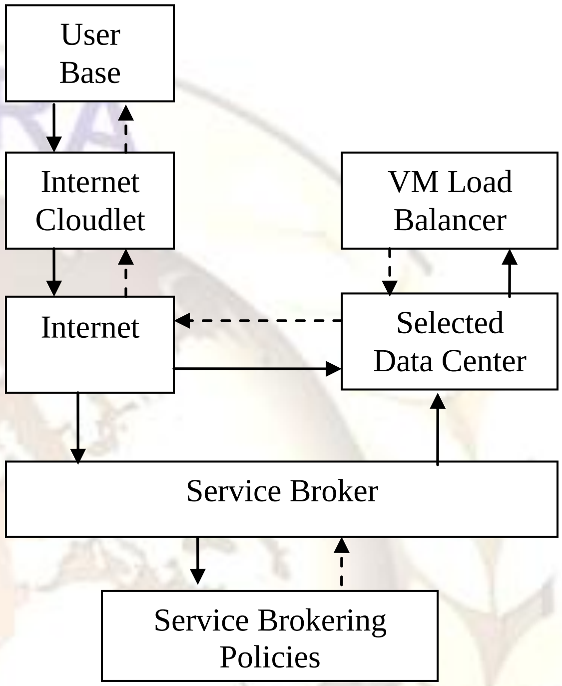

Fig 2: Servicing user requests.

V. Service Proximity-Based Routing

Proximity-Based Routing policy enables the service broker to choose a data center that is nearer to the user region from where the user request is originated without considering other parameters such as cost, response time, etc., If more than one data center is present in the same region, the data center is selected randomly. At times, this mode of selection may lead to overloading of the closest data center. With overloading, there is every possibility of SLA violation.

VI. Enhanced Service Broker Policy

I propose an enhanced proximity-based routing policy that avoids the overloading of the nearest data center by routing the traffic to the neighboring data center in the same region with improved response time; thereby, reducing the chances of SLA violation and enhancing the response time of the data center.

VII. Simulation Configurations and Results

Case 1: I have created two data centers (DC1 and DC2) with the following configuration. I have created one user base (UB1) with the following properties. The selected service brokering policy for this simulation is proximity-based routing. The simulation is run for 24 hours.

Table 1: Application Development Configuration

Data Center	No of VM	Memory	Bandwidth
DC1	5	512	1000
DC2	5	512	1000

Table 2: Data Center Configuration

Name	Region	OS	VMM	Physical HW Units
DC1	2	Linux	Xen	2
DC2	2	Linux	Xen	1

Table 3: User Bases

Name	Region	Req /User /Hr	Avg Peak Users	Avg Off Peak Users
UB1	2	60	1000	100

User Grouping Factor in User Base: 100 Request Grouping Factor: 100 Executable instruction length per request (bytes): 250
Load Balancing Policy across VMs in a Single DC: Throttled

Table 4: Response Time

DataCenter	Avg ms	Min ms	Max ms
DC1	2.60	0.03	4.01
DC2	3.11	0.04	5.03

Case 2: Simulation configuration is identical to the case 1 except that I have created second data center DC2 with 4 hardware units. In the proposed policy, I have made service broker to select next neighboring data center with high configuration (not randomly) in the same region so as to avoid overloading of the original data center.

Table 5: Response Time

DataCenter	Avg ms	Min ms	Max ms
DC1	2.59	0.03	4.01
DC2	2.58	0.03	4.30

The simulation results show that the response time has improved considerably by 17% when I compare it with the Table 4 data.

VIII. Concluding Remarks

I have proposed a 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. This policy is capable of reducing the SLA violation.

References

[1]. Bhathiya Wickremasinghe, Rodrigo N. Calheiros, and Rajkumar Buyya, “CloudAnalyst: A CloudSim-based Visual Modeller for Analysing Cloud Computing Environment and Applications”

[2] L. Kleinrock. A Vision for the Internet. ST Journal of Research, 2(1):4-5, Nov. 2005.

[3] R. Buyya, R. Ranjan, Anton Beloglazov, César A. F. De Rose, and R. N. Calheiros, “CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms”