RESEARCH ARTICLE OPEN ACCESS

An Efficient Service Broker Policy for Cloud Computing Environment

Kunal Kishor[1], Vivek Thapar[2]
Research Scholar[1], Assistant Professor[2]
Department of Computer Science and Engineering,
Guru Nanak Dev Engineering College
Punjab Technical University
Ludhiana, 141006
Punjab –India

ABSTRACT

As almost all the applications are being provided over the internet, the need for computing resources is shifting from the user's location to the service provider. The concept of services has gained popularity with the widespread use of the term "cloud computing", which is a new paradigm that has been defined to address user requests on a pay-per-use basis. With the greatest benefit being elasticity in terms of increase or decrease of computing resources like computation power, storage and bandwidth, cloud is providing better computing solutions to the users of its services. But the success of these solutions lies in the use of efficient policies and algorithms that govern the underlying concept of cloud computing. These policies involve service brokerage, load balancing, virtual machine management and service level agreements. The broker is an intermediary between the client and the cloud service provider and hence the role of a broker is quite significant. Service broker policies are used by a service broker of a cloud solution in order to decide the data center to which the requests have to be routed. Testing such policies in different situations is very necessary for the best results. But carrying out this analysis for a large architectural setup in a real environment is not feasible. Therefore, we make use of simulation tools, which provide a modeling environment and provisioning facilities for analyzing the new algorithms in a successful manner. Cloud Analyst is a graphical cloud simulation tool that provides the necessary simulation environment for executing and analyzing various cloud scenarios. It also provides facilities to implement new policies and algorithms. The proposed broker policy makes use of a proportion weight to decide which data center has to be selected for servicing a particular user request. The hardware configuration of the data centers remains the same; only the number of machines available is different in each of the data centers. Therefore, the use of a proportion that tells how much workload that particular data center can handle is used to assign a comparable proportion of cloudlets to that data center.

Keywords:- Cloud computing, Software as a Service, Platform as a Service, Hardware as a Service, Hypervisor, Service proximity, Performance optimized routing, Dynamically reconfigurable routing, Round Robin, Throttled, Active monitoring.

I. INTRODUCTION

Cloud computing has gained popularity and pace recently, through the provision of well defined services in almost every field. The computing power that was once a need for the client premises is now moving to the cloud service providers, with clients just making use of a simple terminal with a web browser. But the power and user of the application has not diminished. Indeed, the quality of service has improved by shifting the burden to the service provider. The cloud computing model has proved to be a boon for end users, system administrators, software developers and even the IT buyers, corporate and federal clients. These users have been drawn by this paradigm because of the features it provides like user self-provisioning, low operational cost and freedom from capital cost, availability of a large pool of resources, multi-tenant user access, reliability, security and measured service. The basic principle that governs the cloud computing paradigm is the provision of computing resources like computing power, storage and bandwidth in the form of "services" on a pay-per-use basis. The services are provided in the form of Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS), which may be deployed in public, private, community or hybrid form. The main feature that characterizes cloud computing is the use of virtualization (1). Virtual machines are run over the available hardware to address the user needs. So, virtual machine management forms an important part of this concept, and is performed by a dedicated part called the hypervisor. The selection of virtual machines whenever the workload is encountered is done by the load balancer, whose aim is to distribute the load in such a way that no virtual machine is flooded by requests at one time, while remaining idle at other times. Above this level lies another abstraction called the service broker, which is the intermediary between the users of the cloud and the cloud service providers. It makes use of the existing service broker policies in order to rote the user

request to the most appropriate data centre. Therefore, the selection of the best policy decides the response time of a particular request and the efficiency of utilization of the data centre. Data centres are owned and managed by the service providers at distinct locations, and based on utilization, a service provider may decide to configure its data centres with different types of hardware. Also, the hardware keeps on changing time to time according to the user requirement. Therefore, even if service providers try to maintain uniformity in the selection of hardware, they have to increase or decrease the number of machines as per the needs of clients. The existing broker policies do not base their selection of data centres on the internal hardware configurations, but on the location of data centres, response time or current execution load. We propose a broker policy that selects a data centre for fulfilling its requests based on the proportion of hardware available, even when the hardware available at different data centres is of different configuration. Section II discusses the existing service broker policies and the related work. Section III provides the problem formulation. Section IV proposes the new broker policy. Section V discusses the results of simulation using the new policy. Section VI concludes the topic and provides an idea on the future scope.

II. BACKGROUND

The entire process of serving a client is a part of any one of the services defined in the service model. It begins with a request for a particular resource or application, be it for development, or just accessing the storage of the service provider. The request is serviced by the cloud service provider through a series of steps, the first one passing through a cloud service broker, which acts as the intermediary between a cloud consumer and the cloud service providers. The service broker makes use of any one of the available service broker policies in order to send the request to the most appropriate data centre. The role of a service broker is shown in figure 1.

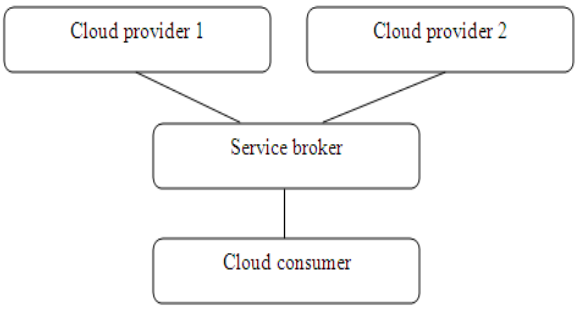

Figure 1: Role of service broker

After choosing the data centre that is going to perform computation, the load balancer at the data centre comes into action. It makes use of the implemented load balancing algorithms (2) to select the appropriate virtual machine to which the request has to be sent for execution. The innermost abstraction layer comprises virtual machine management. The hypervisor (1) or the virtual machine manager is responsible for the management and migration of virtual machines in the cloud data centres.

Out of the above tasks, the use of an efficient service broker policy is quite necessary to ensure that the later tasks are carried out with efficiency and least response time. Literature shows that quite a lot of research has been done in this regards. The three most frequently used service broker policies (3) are explained below:

A. ***Closest data centre policy:*** This policy makes use of the concept of region proximity in selecting the data centre to which the user request has to be guided. A region proximity list is maintained by making use of the "lowest network latency first" criteria to set the order of occurrence of data centers in the list. The data centre that occurs first in the list, i.e., the closest data centre is selected to fulfill the request using this policy. In case more than one data centers with the same latency are available, a random selection of the data centers is made. This policy is therefore beneficial in case the request can be satisfied by a data center that is quite close or within the same region.

B. ***Optimal response time policy:*** This service broker first identifies the closest data centre by making use of the network latency parameter, as in the previous policy. Then, the current response time is estimated for each of them. If the estimated response time is the one for the closest data centre, then the closest data centre is selected. Otherwise, the closest data centre or the data centre with the least response time is selected with a 50:50 chance of occurrence.

C. ***Dynamically reconfigurable routing with load balancing:*** This broker policy (4) makes use of the current execution load in order to scale the application deployment. It also increases or decreases the number of virtual machines accordingly. The router needs to hold an additional responsibility of scaling the application deployment. This is done based on the load that it is currently facing. Scaling is done in this policy by considering the current processing times and the best processing time ever achieved.

Sandhu and Kaur (4) in their work titled *"Modelling Local Broker Policy Based on Workload Profile in Network Cloud"*, explained that workload based applications are handled using two scheduling policies: Random Non-overlap and Workload profile. They compared these two scheduling policies based on three parameters: Execution (mean) time, Response (mean) time and Waiting (mean) time. A methodology was used wherein a cloud user base was initially created and later the local and global brokers were created. Then the cloudlets were submitted to the data centre after creating a workload scheduling policy. The comparison shows that Workload based profile policy is better when compared to Random non overlap policy in each of the parametric values.

Chudasama et al. (5) , in their paper "Cost effective selection of Data center by Proximity-Based Routing Policy for Service Brokering in Cloud Environment", proposed an enhanced proximity-based routing policy for cost-effective selection of data centers, which just made use of the region proximity list in the existing policy.

Dash et al., (6) in their paper "Cost Effective Selection of Data Center in Cloud Environment", used the cloud analyst simulation tool to compare the performance of the existing data center selection policies. Throttled load balancing algorithm was used to simulate the execution environment. They also compared the results with the Round Robin load balancing algorithm in order to estimate the response time and the processing time.

III. PROBLEM FORMULATION

Cloud computing today has now been rising as new business model. The increasing cloud services offer the great opportunity for clients as well as the cloud service providers to maximize their benefits, which however raises the new challenges on how to improve these technologies.

In cloud computing to gain high user satisfaction and increased resource utilization, equitable and proper allocation of every computing resource is essential. Various techniques are being used for the same with their added advantages and disadvantages. One of the algorithms used for proper allocation of data center to the requests is service proximity based routing, which selects the nearest data center available on that region. There are disadvantages appended to it. As the data center is randomly selected when there are more than one data center in the same region so, there can be the possibility of different selection each time for the same configuration whose results can be difficult to interpret for developers/researchers. The possibility of higher cost is also a threat in such random selection.

While using such method, data center selection is done in round robin manner, so the distribution of request is done uniformly. As efficiency of the underlying data center is ignored i.e. irrespective of its request processing capacity the allocation of requests are done. So the utilization of data center resources cannot be done effectively. There may be a situation when a data center with huge capacity of processing the request got the less number of cloudlet for execution and a data center with less capacity got the more number of cloudlet. So data center processing time may be differ.

To overcome the demerit of random selection of data center in region with more than one data center are available, an efficient and effective method is proposed for the selection of data center based on efficiency to handle the request. The stress has been laid on developing an efficient algorithm with improved attributes of cloud computing environment. The proposed algorithm will be compared with existing service broker algorithms that will let to give better results.

IV. PROPOSED ALGORITHM

The proposed algorithm is based on the service proximity service broker policy to select the data center. The proposed algorithm makes use of proportion weights in order to provide an insight into the computation power of the data center. Different data centers may be of the same hardware configuration but they often contain physical machines in varied number. So the data center with more number of physical machine have the capability the process more amount of resource with respect to the data center within which number of physical machine is less. In the service proximity based algorithm the resource allocation is done uniformly by ignoring the underlying infrastructure of data center. As for example, if more than one data centers are present in a region, one of these data centers may contain one physical machine while the other may contain three physical machines. Though the hardware configuration is the same, but the data centers can handle different amount of workloads because of the provision of larger number of computing elements in the second data center. The proposed strategy therefore, makes use of this logic to assign workloads to the data centers to consider the underlying infrastructure of the data center. A proportion weight is assigned to both the data centers according to the number of computing elements available. In above mentioned case, the first data center is assigned a proportion of 1 and the second one is assigned a proportion of 3. This is used to direct the cloudlets to the data centres in the specified way, i.e., for every burst of cloudlets, the selection of data centres by the ratio of 1:3. This is bound to improve resource utilization of the data centre as well as the disadvantage of random selection and also make a great improvement in overall response time and data center processing time.

The key steps of the proposed algorithm is summarised below:-

i. Calculate the number of data centres present in the region for which the simulation is to be performed.
ii. If only one data centre is present then go to step (vi), else go to the next step for further execution.
iii. For multiple data centres in that region, calculate the resource handling capacity of each data centre.
iv. Assign a proportion weight to the data centre according to the underlying infrastructure to handle the resource.
v. Select the data centre in a circular fashion, followed by the number of resources allocated to each data centre according to their proportion weight.
vi. Send the request to the selected data centre.
vii. Analyse the result of simulation.

This algorithm is implemented using the Cloud Analyst simulation toolkit, to analyze the result graphically. The implemented algorithm is modified with respect to the service proximity algorithm. The changes are done on data center

selection process and how the selection process is done according to the proportion weight of the data center. The pseudo code of the proposed algorithm is mentioned below.

Pseudo code 1: (Data Center Selection)
Input: Region number
Output: Destination DC name
Function: getAnyDataCenter(region)
regionalList ← regionalDataCenterIndex.get(region)
if regionalList is not NULL **then**
listSize ← regionalList.size();
if listsize == 1 **then**
dcName ← regionalList.get(0)
else
Assign the weight to the entire data center
dataCenterID ← **proportionalDCSelection(listSize)**
dcName ← regionalList.get(dataCenterID);
end if
end if
return dcName
end function
In the above pseudo code, data centers are assigned for resource allocation. Resource allocation using the proposed policy is discussed below. The pseudo code is written for convenience, assuming that there are three data centers available in the region.

Pseudo code 2: (Algorithm with respect to proportion selection)
Input: listsize, Data Center Detail
Output: Destination Data center ID
Function: proportionalDCSelection(listSize) // if list size is 3 as there are 3 data center
x0 ← Proportion Weight of 1st DC;
x1 ← x0 + Proportion Weight of 2nd DC;
x2 ← x1 + Proportion Weight of 3rd DC;
i++; // (we are taking I as a static variable which is incremented with every call of the function)
if (i < x0)
datacenterID ← datacenterID of 1st datacenter
else if (i > x0-1 && i < x1)
datacenterID ← datacenterID of 2nd DataCenter
else if (i > x1-1 && i < x2)
datacenterID← datacenterID of 3rd DataCenter
else i ← 0
Return datacenterID ;
End Function

The proposed algorithm aims to show the effectiveness of proportion weight to select the data center in order to achieve better resource utilization and remove the disadvantages of random selection.

V. RESULTS AND DISCUSSION

The proposed policy has been implemented in the Cloud Analyst simulation tool, which is based on the CloudSim (7) extensible simulation toolkit.

The simulation configuration is as:

Parameter	Value Used
UB Name	UB1
Region	2
Request Per User Per Hour	60
Data Size Per Request	100
Peak hour start(GMT)	3
Peak hour end (GMT)	9
Avg Peak Users	400000
Avg Off Peak Users	40000
DC 1 – No Of VM	80
DC 2 – No Of VM	40
DC 3 – No Of VM	20
VM Image Size	10000 MB
VM Memory	1024 MB
VM Bandwidth	1000 bps
DC 1 – No Of Physical Machine	20
DC 2 – No Of Physical Machine	10
DC 3 – No Of Physical Machine	5
DC – Memory Per Machine	2048 MB
DC – Storage Per Machine	40000 MB
DC – Available BW Per Machine	4000bps
DC – No Of Processors Per Machine	4
DC – Processor Speed	1000 MIPS
DC – VM Policy	Time Shared
User Grouping Factor	10000
Request Grouping Factor	1000
Executable Instruction Length	250
Load Balancing Policy	Throttled

Table 1: The Cloud Analyst Configuration for simulation

The entry for number of virtual machines, as mentioned in the table 1 for DC 1, DC 2 and DC 3 is 80, 40 and 20. So the considerable proportion weight is in the ratio 4:2:1. Therefore, using the proposed algorithm, DC 1 is assigned to process the first 4 cloudlet, DC 2 is assigned to process the next 2 cloudlet and DC 3 is assigned to process the last cloudlet, out of 7 first cloudlets and the whole selection process is done in a repeated manner to process the entire set of workloads.
Simulations run using the same configuration for each of the two existing and the proposed broker policy reveal an improvement in the results. The results include an overall response time summary, response time by region and data centre request servicing time. The throttled load balancing policy is being used to manage the load across the VM in the data centers .The summary of the results from the three simulations with three service broker policies: closest data

centre, Optimized Response Time and the Proposed Proportion Based Algorithm is shown in table 2 below:

Table 2: Results for overall response time and DC processing time

	Closest Data Center	Optimize Response Time	Proposed algorithm
Over All Response Time	594.60	616.00	481.54
DC processing time	536.91	558.19	423.52

The table clearly shows that the values of overall response time and DC processing time for the proposed algorithm is far better than that for the other two existing broker policies. Graph for the same is shown in figure 2 below:

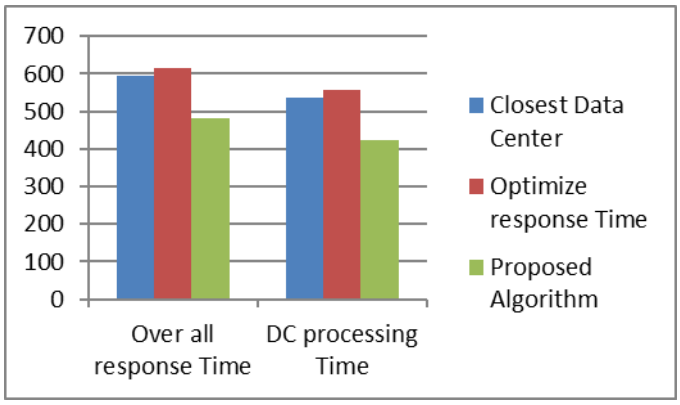

Figure 2: Bar chart for overall response time and DC processing time for the three broker policies

Table 3 shows the processing times of data centers individually for each of the three broker policies.

Table 3: Result of individual data centre processing time

	Closest Data Center	Optimize Response Time	Proposed Algorithm
DC 1 Processing Time	372.62	439.27	423.40
DC 2 Processing Time	478.64	654.64	423.33
DC 3 Processing Time	810.40	615.91	423.37

Table 3 reveal the DC processing times of all the data centres using the three policies. If we consider the closest data center, the gap of processing time among all the data center is much more. When we consider the optimize response time as service broker the gap of processing time is also available. The data center with high capacity (DC1) finishes their task earlier and remains ideal and data center with low processing capacity takes more time accordingly. When the proportion based broker policy is being considered the values have greatly improved for DC 2 and DC 3. The workload of DC 2 and DC 3 is shifted to DC 1, whose capacity to process the request is more than the others. The DC processing time for DC 1, DC 2 and DC 3 are quite comparable when using the proposed policy, which show a better utilization of all the data centre. Graph for the same is shown in figure 2 below, which is showing the comparable processing time of proposed policies.

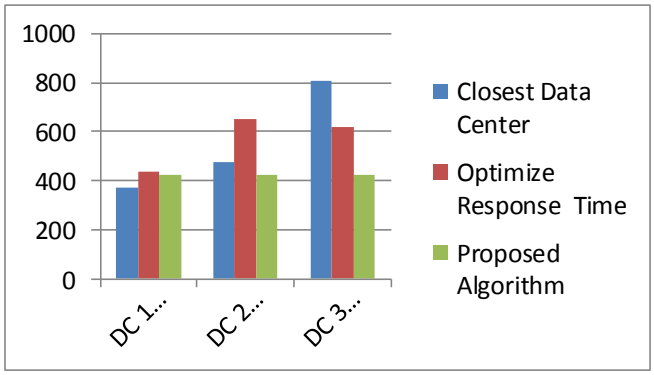

Figure 3: Bar chart for individual DC processing time

VI. CONCLUSION AND FUTURE SCOPE

The major advantage of cloud computing is the availability of computing and storage resources in the form of various services, on a pay-per-use basis. Also, the clients are free from the worry of knowing about the underlying hardware that is servicing their requests. In order to maintain this feature, it is necessary to efficiently utilize the available resources. Response time is the major governing factor when considered from the point of view of the user. A better response time and data centre request servicing time account for service efficiency, and the dependent directly on the choice of an appropriate data centre. If the correct data centre in terms of data processing time and response time is chosen, the request can successfully be serviced, without any delays. The proposed broker policy is a move in this direction of perfection. It makes use of the concept of proportion weights in order to assign the workload to data centers. Whereas the existing broker policies based their decisions on the location or randomly or optimality of data centers, the proposed policy takes into consideration the efficiency of underlying hardware, where a greater number of hardware machines (though of the same configuration) mean greater number of virtual machines can be created on that underlying hardware configuration and hence a larger number of cloudlets that they can handle by that data center. With implementation of proposed data center selection policy, the resource utilization of data center can be done effectively. Also we got a great improvement in the overall response time, the regional request servicing time and the data centre processing time using the proposed service broker policy.

The proposed algorithm considers different data centers that are of the same configuration. The future work in this regards can be done by maintaining a proportion for data

centres in such a way that distinct hardware configurations are also considered.

REFERENCES

[1] VMWare. Virtualization overview. White paper. Palo Alto : VMWare. p. 11.

[2] A Survey of Load Balancing Algorithms in Cloud Computing Al. Brar, Harmandeep Singh, Thapar, Vivek and Kishor, Kunal. 3, Ludhiana : s.n., June 2014, International Journal of Computer Science Trends and Technology, Vol. 2, p. 4. ISSN: 2347-8578.

[3] Wickremasinghe, Bhathiya. CloudAnalyst: A CloudSim-based Tool for Modelling and Analysis of Large Scale Cloud Computing Environments. CSSE department, University of Melbourne. Melbourne : s.n., 2009. p. 44, MEDC Project Report.

[4] Modeling Local Broker Policy Based on Workload Profile in Network Cloud. Sandhu, Amandeep and Kaur, Maninder. 8, Banur : s.n., August 2013, International Journal of Science and Research, Vol. 2, p. 5. ISSN: 2319-7064.

[5] Cost effective selection of Data center by Proximity-Based Routing Policy for Service Brokering in Cloud Environment. Chudasama, Devyaniba, Trivedi, Naimisha and Sinha, Richa. 6, 2012, International Journal of Computer Technology & Applications, Vol. 3. ISSN:2229-6093.

[6] Cost Effective Selection of Data Center in Cloud Environment. Dash, Manoranjan, Mahapatra, Amitav and Chakraborty, N R. 1, 2013, International Journal on Advanced Computer Theory and Engineering, Vol. 2. 2319 – 2526.