Service Broker Algorithms in Cloud Computing

Divya Raj K , Nijeesha G, Santhosh B
Dept of MCA, AIMIT,
St Aloysius College(Autonomous), Mangalore

***Abstract* - A virtual pool of resources is provided to the users by means of Cloud Computing. Efficient use of the limited resources available is one of the requirements in the cloud In this paper we have compared the various service broker algorithms such as Service Proximity Based Routing, Performance Optimized Routing, Enhanced Proximity-Based Routing Algorithm and Resource Location and Cost based Performance Optimized Routing algorithms**

Key words: cloud computing, scheduling, service broker algorithms

1. INTRODUCTION

Cloud computing provides dynamic, virtualized and scalable on demand resources over the internet. This also helps the users to access various applications as well as the related data items from anywhere. Various companies could rent storage as well as computational resources via cloud and thus reducing their infrastructure cost. Cloud computing provides a pay as you go model which allows the companies to make use of software, storage, platforms as well as infrastructure.

1.1 Introduction to Scheduling

Allocation of jobs to virtual machines is done by the scheduler. It is used to allocate the jobs to the various resources. There are various constraints given by the users. These can be deadlines such as jobs to be completed within a specified amount of time as well as limiting the budget. There are also certain constraints that are specified by the service providers so as to maximize the utilization of resources and benefits.

1.2 Optimization Criteria

Optimization criteria will specify the goals of the scheduling process and is used when making decisions with respect to scheduling. We have to increase the usage of resources, increase the number of jobs successfully completed as well as have to minimize the response time. A constraint and a criterion can be used in defining the objective function. Therefore the value of the objective function could be maximized or minimized based on the criteria that is mentioned by the user.

2. Introduction to Cloud Application Service Broker

Service Broker controls the traffic between the User Bases and the Data Centres. This will determine which Data Centre will service the jobs from the User Base.

2.1 Service Broker Algorithms

2.1.1 Service Proximity Based Routing[2]

This is the simplest Service Broker implementation.

1. ServiceProximityServiceBroker 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 ServiceProximityServiceBroker for the destination DataCenterController.
3. The ServiceProximityServiceBroker retrieves the region of the sender of the request and queries for the region proximity list for that region from the InternetCharacteristics. This list orders the remaining regions in the order of lowest network latency first when calculated from the given region.
4. The ServiceProximityServiceBroker 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.

2.1.2 Performance Optimized Routing[2]

This policy is implemented by the BestResponseTimeServiceBroker, which extends the ServiceProximityServiceBroker.

1. BestResponseTimeServiceBroker maintains an index of all Data Centers available.
2. When the Internet receives a message from a user base it queries the BestResponseTimeServiceBroker for the destination DataCenterController.
3. The BestResponseTimeServiceBroker identifies the closest (in terms of latency) data center using the ServiceProximityServiceBroker algorithm.
4. Then the BestResponseTimeServiceBroker iterates through the list of all data centers and estimates the current response time at each data center by

a. Querying the last recorded processing time from InternetCharacteristics.

b. If this time is recorded before a predefined threshold, the processing time for that data center is reset to 0. This means the data center has been idle for a duration of at least the threshold time.

c. The network delay from InternetCharacteristics is added to the value arrived at by above steps.

5. If the least estimated response time is for the closest data center, the BestResponseTimeServiceBroker selects the closest data center.
Else, BestResponseTimeServiceBroker picks either the closest data center or the data center with the least response time with a 50:50 chance (i.e. load balanced 50:50).

2.1.3. Enhanced Proximity-Based Routing Algorithm [14]

1. ServiceProximityServiceBroker 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 ServiceProximity ServiceBroker for the destination DataCenterController.
3. The ServiceProximityServiceBroker retrieves the region of the sender of the request and queries for the region proximity list for that region from the InternetCharacteristics.This list orders the remaining regions in the order of lowest network latency first when calculated from the given region.
4. The ServiceProximityServiceBroker picks the first data center located at the earliest/highest region in the proximity list and satisfied by SLA. If more than one data center is located in a region, select the datacenter from the region where Vm$/Hr is minimum. When there are multiple datacenters with the same Vm$/Hr is present , select the datacenter randomly.

2.1.4. Resource Location and Cost based Performance Optimized Routing [13]

i. BestResponseTimeServiceBroker maintains an index of all Data Centers based on the user SLA, by default it maintains the index of all Data Centers
ii. When the Internet receives a message from a user base it queries the BestResponseTimeServiceBroker for the destination DataCenterController.
iii. The BestResponseTimeServiceBroker identifies the closest (in terms of latency) data center using the ServiceProximityServiceBroker algorithm and selects data center that are listed in SLA

iv. Then the BestResponseTimeServiceBroker iterates through the list of all data centers and estimates the current response time at each data center by

a. Querying the last recorded processing time from InternetCharacteristics.

b. If this time is recorded before a predefined threshold, the processing time for that data center is reset to 0. This means the data center has been idle for a duration of at least the threshold time.

c. The network delay from InternetCharacteristics is added to the value arrived at by above steps.

v. If the least estimated response time is for the closest data center, the BestResponseTimeServiceBroker selects the closest data center. Else selects datacenter with the least response time.
When there are multiple datacenters are available it selects the datacenter where the processing cost is minimum.

3. Simulation results and analysis

The algorithms are simulated in a simulation toolkit CloudAnalyst[2]

3.1 Introduction to Cloud Analyst

To support the infrastructure and application-level requirements such as modelling of on demand virtualization arising from Cloud computing paradigm, enabled resource simulators are required. Few simulators like CloudSim [1] and CloudAnalyst [2] are available. CloudAnalyst has been used in our paper as a simulation tool. A snapshot of the GUI of CloudAnalyst simulation toolkit is shown in figure 1(a) and its architecture is shown figure 1(b).

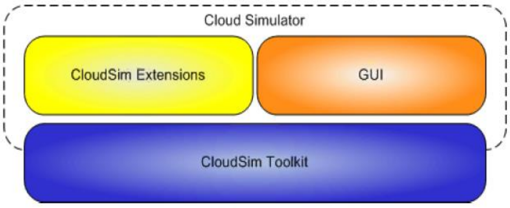

Figure 1. CloudAnalyst built on top of CloudSim toolkit

3.2 Simulation scenario1

In this scenario User Base and data center configurations are as given in Table 1 and Table 2 and it is assumed that UB1 user requests can be executed in any datacenter except in Region 1. In the Proximity-Based Routing algorithm always selects closest data center and most of the cases data center within the user base.

S.No	User Base	Region	Online users during peak hrs.	Online users during off-peak
1	UB1	1	1000	100

Table 1 : Configuration of User Bases

Name	Region	Arch	OS	VMM	Cost per VM $/Hr	Memory Cost $/s	Storage Cost $/s	Data Transfer Cost 4/Gb
DC1	1	x86	Linux	Xen	2	0.05	0.1	0.1
DC2	1	x86	Linux	Xen	1	0.05	0.1	0.1
DC3	2	x86	Linux	Xen	1	0.05	0.1	0.1
DC4	2	x86	Linux	Xen	1	0.05	0.1	0.1
DC5	4	x86	Linux	Xen	1	0.05	0.1	0.1

Table 2 : Configuration of Data Centers:

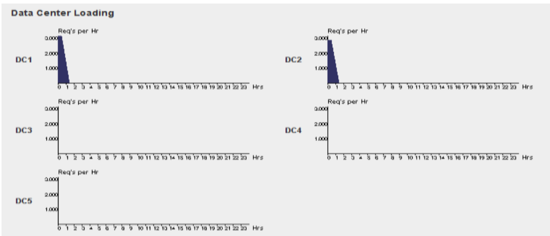

Fig 4: Service Proximity based routing algorithm- Datacenter Loading

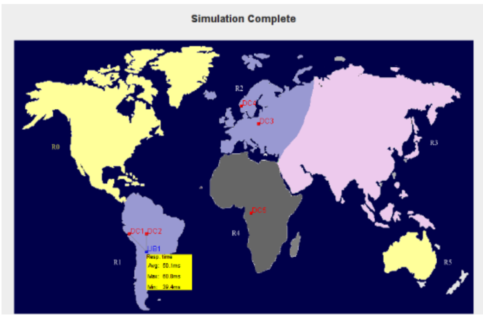

Fig 5: Service Proximity based routing algorithm

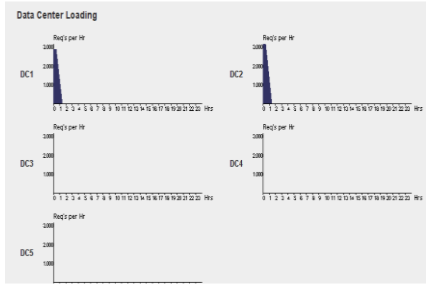

Fig 6: Performance Optimized Routing- Datacenter Loading

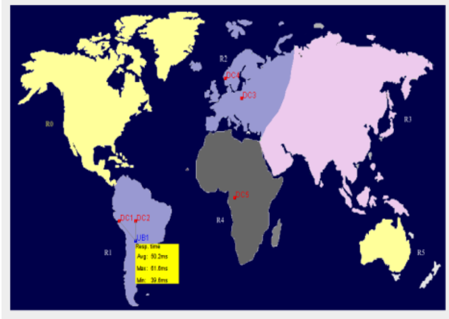

Fig 7: Performance Optimized Routing

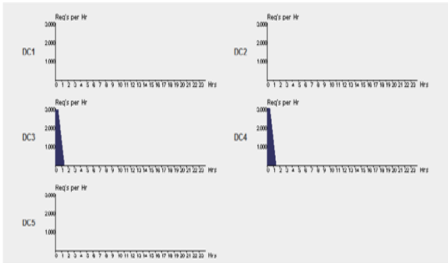

Fig 8: Resource Location and Cost based Performance Optimized Routing-Datacenter Loading

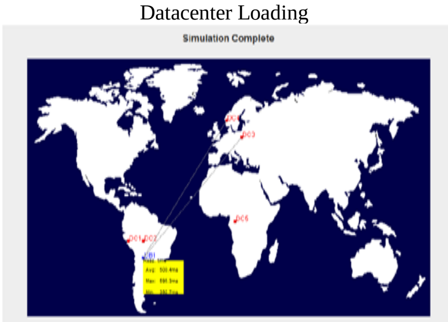

Fig 9: Resource Location and Cost based Performance Optimized Routing

Fig 4 and Fig 6 shows the results of datacenter loading .It shows the loads are equally distributed between the datacenters.. Here UB1 region is 1. Results shows that UB1 requests are processed in data centers DC1 and DC2 .The assignment of the user bases to the data centers also depicted in the final simulation results as shown in Fig 5 and fig 7. These experiments shows that Service Proximity based routing algorithm always it selects the closest datacenter and performance optimised algorithm selects If least respose time data center is the closest data center then it selects the closest data center else selects closest data center and datacenter with the least response time with a 50:50 chance.

Fig 8 shows the results of data center loading . Here UB1 region is 1 and the SLA requirement is request can be executed in any region other than region1. Results shows that

UB1 requests are processed in data centers DC3 and DC4 . It satisfies SLA requirement . The data centers DC3 and DC4 are in region 2 and cost per VM$/Hr is same , the Fig 8 is also shows that load is equally distributed between the data centers. The assignment of the user bases to the data centers also depicted the final simulation results as shown in Fig 9.

3.2 Simulation scenario2

In this scenario User Base and data center configurations as given in Table 3 and Table 4 and It is assumed that UB1 user requests can be executed in any datacenter except in region 1 and region 3. UB2 user requests can be executed in any datacenter except in region 2 and 4 and UB3 can be executed in any datacenter.

S.No	User Base	Region	Online users during peak hrs.	Online Users during off-peak
1	UB1	1	1000	100
2	UB2	2	1000	100
3	UB3	3	1000	100

Table 3 : Configuration of User Bases

Name	Region	Arch	OS	VM	Cost per VM $/Hr	Memory Cost $/s	Storage Cost 4/s	Data Transfer Cost 4/Gb
DC1	1	x86	Linux	Xen	2	0.05	0.1	0.1
DC2	1	x86	Linux	Xen	1	0.05	0.1	0.1
DC3	2	x86	Linux	Xen	1	0.05	0.1	0.1
DC4	2	x86	Linux	Xen	1	0.05	0.1	0.1
DC5	4	x86	Linux	Xen	1	0.05	0.1	0.1

Table 4 : Configuration of Data Centers:

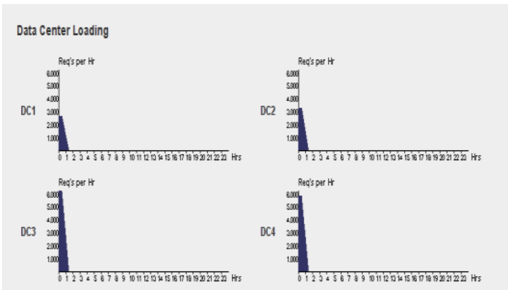

Fig 10: Service Proximity based routing algorithm- Datacenter Loading

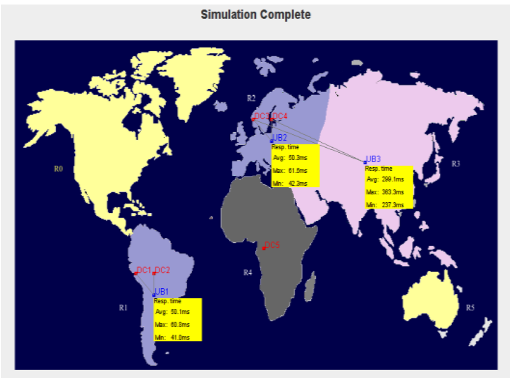

Fig 11: Service Proximity based routing algorithm

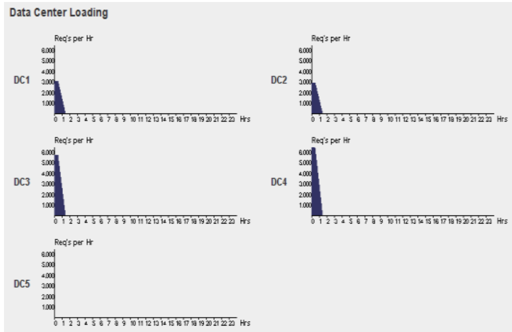

Fig 12 : Performance Optimized Routing- Datacenter Loading

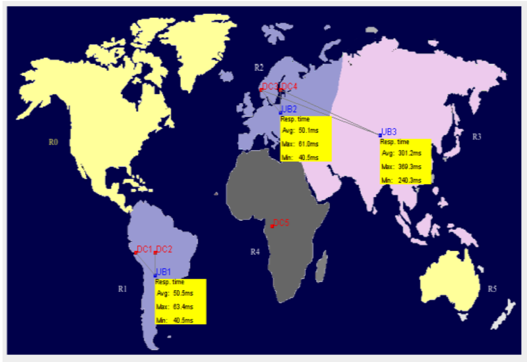

Fig 13: Performance Optimized Routing

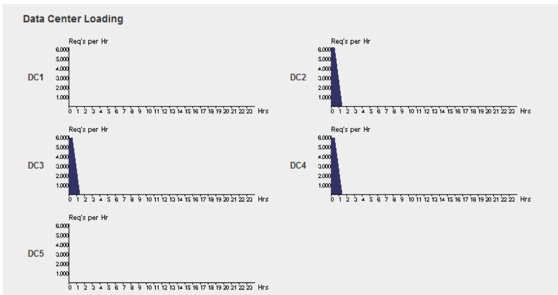

Fig 14: Resource Location and Cost based Performance Optimized Routing- Datacenter Loading

Fig 15Resource Location and Cost based Performance Optimized Routing

Fig 10 and Fig 12 shows the results of datacenter loading .It shows the loads are equally distributed between the datacenter. The assignment of the user bases to the data centers also depicted in the final simulation results as shown in Fig 11 and fig 13. These experiments shows that Service Proximity based routing algorithm always it selects the closest datacenter and performance optimised algorithm selects If least response time data center is the closest datacenter then it selects the closest datacenter else selects closest data center and datacenter with the least response time with a 50:50 chance.

Fig 14 shows the results of data center loading .Here UB1,UB2,UB3 user bases are in the region 1,2 and 3 respectively. The SLA requirement for UB1 is request can be executed in any region other than 1 and 3, UB2 request can be executed in any region other than 2 and 4 and there is no restriction of data centers for userbase UB3. The results in Fig 14 and Fig 15 shows that the UB1 region is 1 and closest are data centers are DC1 and DC2 , requests are executed in region 3 by the data centers DC3 and DC4 and satisfies the given SLA. The SLA requirement for UB2 is request can be executed in any region other than 2 and 4.The results in Fig 14 and Fig 15 shows that though UB2 region is 2 and closest are data centers are DC3 and DC4 , requests are executed in region 1 by the data center DC1 and DC2 . Here cost of VM$/Hr of data center of DC1 is more than DC2. Therefore majority of the requests are executed in DC2. Since there is no restrictions on the user base UB3 , results shows that it was executed in the closest data centers DC3 and DC4. Here the cost of VM$/Hr of data centers are same and the fig 14 shows that the load is equally distributed between the data centers. The assignment of the user bases to the data centers also depicted the final simulation results as shown in Fig 15

4. CONCLUSION

Scheduling is one of the most important task in cloud computing environment. In this paper we have explained different types of scheduling and analyzed various service broker algorithms and compared its data center loading and assignment of user base requests to datacenter. For assigning the requests the algorithm uses different parameters like closest path, performance and location of the resource etc.

REFERENCES

[1] R.N.Calheiros,,Ranjan,A.Beloglazov,C.Rose, R.Buyya,"Cloudsim:A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms",in *Software:Practice and xperience(SPE)*, Vol:41, No:1,ISSN:0038-0644,Wiley Press,USA,pp:23-50,2011.

[2] B. Wickremasinghe, R.N. Calheiros and R. Buyya, "Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications", in Proc. of *Proceedings of the 24th International Conference on Advanced Information Networking and Applications (AINA 2010)*, Perth, Australia, pp.446-452, 2010.

[3] Chieu T.C., Mohindra A., Karve A.A., Segal A., "Dynamic Scaling of Web Applications in a Virtualized Cloud Computing Environment ," in IEEE International Conference on e-Business Engineering, Dec. 2009, pp. 281-286.

[4] Naidila Sadashiv, S. M Dilip Kumar, "Cluster, Grid and Cloud Computing: A Detailed Comparison," The 6th International Conference on Computer Science & Education (ICCSE 2011) August 3-5, 2011. SuperStar Virgo, Singapore, pp. 477- 482.

[5] Vincent C. Emeakaroha, Ivona Brandic, Michael Maurer, Ivan Breskovic, "SLA-Aware Application Deployment and Resource Allocation in Clouds", 35th IEEE Annual Computer Software and Application Conference Workshops, 2011, pp. 298-303.

[6] V. C. Emeakaroha, I. Brandic, M. Maurer, and S. Dustdar, "Low level metrics to high level SLAs - LoM2HiS framework: Bridging the gap between monitored metrics and SLA parameters in cloud environments," In High Performance Computing and Simulation Conference, pages 48 – 55, Caen, France, 2010

[7] M. Maurer, I. Brandic, V. C. Emeakaroha, and S. Dustdar, "Towards knowledge management in self-adaptable clouds," In 4th International Workshop of Software Engineering for Adaptive Service-Oriented Systems (SEASS"10) , Miami, Florida, USA, 2010.

[8] http://www.asd.gov.au/publications/csocprotect/Cloud_Computing_Security_Considerations.pdf

[9] R. Van den Bossche, K. Vanmechelen, J. Broeckhove, Cost-optimal scheduling in hybrid iaas clouds for deadline constrained workloads, in: IEEE 3rd International Conference on Cloud Computing CLOUD, pp. 228–235.

[10] R. Buyya, C. Yeo, S. Venugopal, J. Broberg, I. Brandic, Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems 25 (2009) 599–616.

[11] Martin Randles, David Lamb, A. Taleb-Bendiab, "A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing", 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, pp. 551-556.

[12] Soumya Ray and Ajanta De Sarkar, "Execution Analysis Of Load Balancing Algorithms In Cloud Computing Environment", International Journal on Cloud Computing: Services and Architecture (IJCCSA), Vol.2, No.5, October 2012

[13] Santhosh B ,Dr D.H Manjaiah "Resource Location and Cost based Performance Optimized Routing algorithm in cloud computing" Second International Conference on Emerging Research in Computing, Information, Communication and Applications, August 2014 published in ELSEVIER proceedings with ISBN 9789351072621

[14] Santhosh B ,Dr D.H Manjaiah "Enhanced Proximity-Based Routinng Policy for ServiceBrokering in Cloud Environment",National Conference on Recent Trends in Computing &Information Technology, August 2014 ISBN N.978-81-929306-2-6