Priority Based Algorithm for Load Balancing on Cloud Data Centers

Mr. Sourav Prajapati
Department of MCA, Bharati Vidyapeeth's
Institute of Management and Information
Technology (BVIMIT), Navi Mumbai
Email: souravprajapati31@gmail.com

Dr. Sambhu Rai
Department of MCA, Bharati Vidyapeeth's
Institute of Management and Information
Technology (BVIMIT), Navi Mumbai
Email: shambhumca1@gmail.com

ABSTRACT

Cloud computing has added a new paradigm for user services which allows accessing IT services on the basis of pay-per-use at any time and any location with the help of internet. Due to flexibility in cloud services which are Information as a service , Platform as a service and Software as a service numerous organizations have shifting their business to the cloud and service providers are establishing more data centres to provide services to users. However, it is important to provide cost-effective tasks execution and proper utilization of resources. Several techniques have been reported in the literature to improve performance and resource use based on load balancing.

Clouds are high configured infrastructure delivers platform, software as service, where in infrastructure gives you a bare metal , and software gives you the environment to run the software .which allows customers to make subscription for their requirements under the pay as you go model These model are widely accepted by cloud service provider like AWS, Google cloud, Heroku and etc. Cloud computing is spreading globally, due to its easy and simple service oriented model.

The numbers of users accessing the cloud are rising day by day just because when you are using any service which is been provided by cloud service provider there rarely any down time and no maintenance. Normally cloud is based on data centers which are powerful to handle large number of users. The reliability of clouds depends on the way it handles the loads, to overcome such problem clouds must be featured with the load balancing mechanism Now Load balancing in cloud computing play a major role helping the clouds to increase their capability, capacity which in turn leads to more powerful and reliable cloud service.

Keywords

Load Balancing, platform as a service(Paas), Infrastructure as a service (Iaas) , Software as a service (Saas) , AWS, ,Google cloud, Heroku , cloud computing

1 INTRODUCTION

The performances of computational system depend on multiple concepts like load balancing , CPU cores ,caching .Load balancer is a system which helps the server not to get overwhelming under extreamly high requests by the users by distributing the load of request among multiple server.Thus making the services work under high number of request. At this moment, the system must be in control and operate in accordance with the fundamental priorities. The interaction with factors and some load balancing algorithm which can be applicable for such factors are studied in the current paper and a priority based algorithm has been proposed.

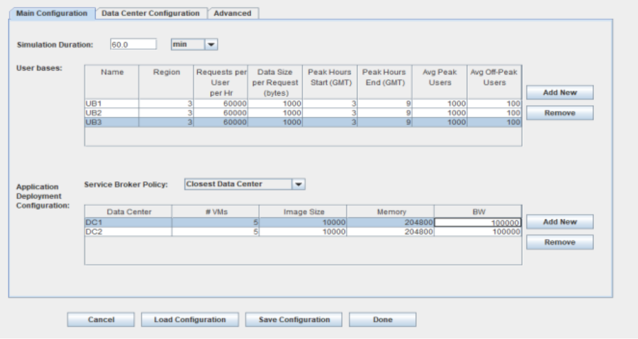
There are multiple types of load balancing algorithms for the improvement and optimization of cloud performances There can be two type of algorithm .dynamic or static, although some algorithms are simple but under some conditions they work more effectively .Cloud computing is a service oriented architecture, which is been provided via internet. The main objective of the multiple service provider is just to maximum resources output and this can be achieved by implementing load balancing algorithm

2 LITERATURE SURVEY

There are six continents in the world are considered as six regions in an application called CloudAnalyst. The user bases and data centers are geographically scattered over the six regions where in we can chose from where the request can come and where the data center would be located. Request from a user base need to be routed to a data center, where these request can be response back to the user. This process decides the efficiency in terms of response time, data center processing time and cost. Service broker plays an important role for achieving these parameters with efficiency.

- Closest data center policy

This policy is service proximity based routing algorithm. As its name suggest, the earliest data center is chosen for servicing the request so that the request can be replied as soon as possible. This proximity list of data centers is prepared in terms of least network latency. When there are multiple closest data centers, one is randomly selected from the proximity list. The optimize response time policy is the performance optimized which optimizes the routing and an its an extension of closest data center policy. Initially, the closest data center is detected. If the response time of the closest data center. There are cases where the data center starts degrading (total latency increases due to traffic) , then the data center with better response time (latency) at that particular time is searched and that data center is been taged as quickest data center.



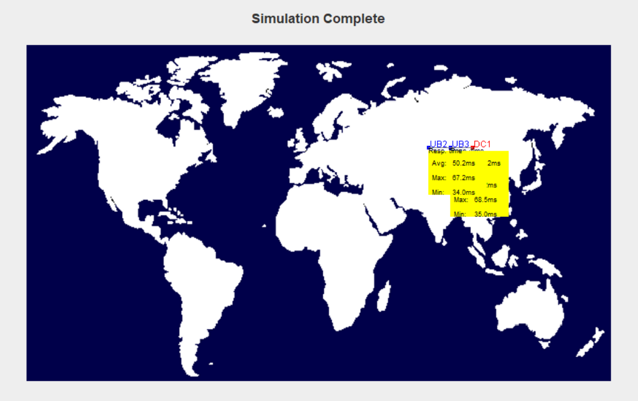
Simulation Duration: 60.0 min

User bases:

Name	Region	Requests per User per Hr	Data Size per Request (bytes)	Peak Hours Start (GMT)	Peak Hours End (GMT)	Avg Peak Users	Avg Off-Peak Users
UB1	3	60000	1000	3	9	1000	100
UB2	3	60000	1000	3	9	1000	100
UB3	3	60000	1000	3	9	1000	100

Application Deployment Configuration: Service Broker Policy: Closest Data Center

Data Center	# VMs	Image Size	Memory	BW
DC1	5	10000	204800	100000
DC2	5	10000	204800	100000



3 PROPOSED ALGORITHM

- Priority based load balancing

The algorithm is based on a simple concept where we serve the user who come from far first thus making them getting more priority from a user (Request) who is near to the server geographically. Thus making the response time better for the user who are far. There are many instances where some user gets response in 5ms even in case where 15ms of response time is fine for the user experience and there are some user who have spends 50ms and need to wait at the server queue so that they can get the response .

Pseudo code

R1 and R1 are the objects of Request which represent user request. Priority method is been called only if two or more request simultaneously arrive at the load balancer assuming these request are users first request since if it is second or nth request buy the user there are chances that a session would have been maintained by the server and if the user is been redirected to any other server it would cause new server to create new session.

```
Function **Priority** ( **Request** R1 , **Request** R2):
If   R1 location > R2 location:
        Increase Priority for R1
        Send it back to server (Request processing server or backend server)
Else:
        Increase Priority for R2
        Send it back to server(Request processing server or backend server)
End:
```

Eg : Suppose there are 5 user request given below with there arrival time and the time to process the request at server side is kept constant so as to determine the total turn around time .

Server is at Mumbai

User Request	Time to reach server	Geo location	Arrival Time	Time to Process at server end
R1	20ms	Delhi	2s	1s
R2	25ms	Nepal	2s	1s
R3	10ms	Goa	3s	1s
R4	3ms	Mumbai	4s	1s
R5	30ms	China	4s	1s

Table 1.1

1.Request R1 and R2 arrive at server and the balancer increases the priority of the R2 where the user request have high travel time because of the geo location thus helping it to decrease the TAT .

2. R1 waits until the servers get free in the request queue and once the server is free it processes the R1 request and R1 have a wait time of 1s as it is the time require for server to process the request.

3. R3 arrive at the server when the server is free so it get wait time of 0s

4. R4 and R5 arrive at the same time and the one who have small time to reach the server will be served first.

5. Every request have a turn around time which is noting but time required for the request to return back the response. We have calculated the

TAT as twice the time to reach the server + waiting time at the server + time required for server to process the request and send back the response

User Request	Time to reach server	Geo location	Arrival Time
R1	20ms	Delhi	2s
R2	25ms	Nepal	2s
R3	10ms	Goa	3s
R4	3ms	Mumbai	4s
R5	30ms	China	4s

Table 1.2

Time to Process at server end	User Request	Time to reach server
1s	R1	20ms
1s	R2	25ms
1s	R3	10ms
1s	R4	3ms
1s	R5	30ms

Table 1.2 (Continue)

4 DISADVANTAGES

1. Increased latency for the user who have geographical advantage.

2. Over head of processing the request at load balancer

3. We need the request to be decrypted before we process it if we have used any encryption.

5 OBJECTIVES

The basic objective of using a load balancer is to distribute the load of user among different server. Load balancer also helps in providing security since the user only interact with the load balance and when the workload is distributed among various servers or network units, even if one node fails the burden can be shifted to another active node. Load balancing lets you evenly distribute network traffic to prevent failure caused by overloading a particular resource.

This strategy improves the performance and availability of applications, websites, databases, and other computing resources. It also helps process user requests quickly and accurately.

6 CONCLUSION

There are may algorithm for load balancing like round robin , least connection , Closest data center policy , and may more and our suggested algorithm priority based all these algorithm are have advantage with respect to you entire architecture . New algorithm which are more real time base can be used with priority based algorithm which is been suggested by the paper to increase its efficiency.

7 ACKNOWLEDGEMENTS

Thankful for getting wonderful opportunity to write this significant paper on the subject of load balancing on cloud data centers. They also assisted us in conducting extensive research, and we are very appreciative of their assistance.

8 REFERENCES

[3] IEEE Paper on International Journal of Advanced Research in Computer Science and Software Engineering https://www.researchgate.net/profile/Parag-kaveri/publication/326508266_Load_Balancing_On_Cloud_Data_Centres/links/5b51b5820f7e9b240ff11c8a/Load-Balancing-On-Cloud-Data-Centres.pdf

[4] Judith Hurwitz, Robin Bloor, and Marcia Kaufman, “Cloud computing for dummies” Wiley Publication [Book].

[5] https://www.xcellhost.cloud/blog on load balancer.

[6] Jasmin James, Dr. Bhupendra Verma “Efficient VM load balancing algorithm for a cloud computing environment” IJCSE, 2012.

[7] Brain Underdahl, Margaret Lewis and Tim mueting “Cloud computing clusters for dummies” Wiley Publication (2010), [Book].

[8] Distributed Computing: Principles, Algorithms, and Systems [Book by Ajay D. Kshemkalyani and Mukesh Singhal]

[9] Understanding Distributed Systems, Second Edition [Book by Roberto Vitillo]