ENHANCED EQUALLY DISTRIBUTED LOAD BALANCING ALGORITHM FOR CLOUD COMPUTING

Shreyas Mulay[1], Sanjay Jain[2]

[1]Student, [2]Assistant Professor Computer Science and Engineering Department,
Amity School of Engineering and Technology (ASET), Amity University Rajasthan, AUR, Rajasthan, India,
shreyasmulay23@gmail.com, jainsanjay17@yahoo.co.in

Abstract

Cloud Computing as the name suggests, it is a style of computing where different users uses the resources on the go i.e. over the Internet. In the recent era, this technology has emerged as a strong option for not only large scale organizations but also for small scale organizations that only access/use the resources what they want. In recent research study, many organizations lose significant part of their revenues in handling the requests given by the clients over the web servers i.e. unable to balance the load for web servers which results in loss of data, delay in time and increased costs. This Paper gives a new enhanced load balancing algorithm by which the performance of their web application can be increased. This Algorithm works on the major drawbacks such as delay in time, response to request ratio etc.

Index Terms: *Cloud Computing, Public Cloud, Load Balancing, Load Balancing Algorithms, Enhanced Equally Distributed Load Balancing Algorithm.*

----------***----------

1. INTRODUCTION

Cloud Computing uses distributed technologies to satisfy the user needs. Cloud Services includes the sharing of resources, delivery of software, infrastructure and storage over the internet based on their demands. The main functions of cloud computing are reduced cost, better performance and satisfy the needs of user at a great extent. Now taking all these functions into consideration web servers are designed which can give the best performance but many times the performance drops drastically why? The answer for this question is the balancing of the load on the servers appropriately by some mechanism which will improve the performance of total system. The Simple logic behind the balancing the load over the servers (nodes) is that distribute total load in a systematic manner i.e. balancing the load on the overloaded node to under loaded node so that the response time from the server will decrease and performance of the servers increased. These algorithms does not take the previous state or behavior into consideration, it depends on the current state of the system because of its dynamic behavior. Some Algorithms works on circular order by handling the process without any priority but enhanced equally distributed load balancing algorithm handles the requests on priority.

2. NEED OF LOAD BALANCING IN CLOUD COMPUTING

Load balancing in clouds is a mechanism that spreads the excess dynamic workload evenly across all the Servers (Virtual Machines). It is used to achieve a high user satisfaction and resource utilization ratio, making sure that no single node is overloaded or under loaded, hence improving the overall performance of the system. Proper load balancing can help in utilizing the availability of given resources, thereby minimizing the resource consumption. It also helps in implementing fail-over (fault tolerance), enabling scalability, reducing response time, time delay, reduces cost etc.

3. EXISTING LOAD BALANCING ALGORITHMS / TECHNIQUES IN CLOUD COMPUTING

There are many load balancing techniques given by the researchers over time to time some have advantages over other and vice versa. Load Balancing is required to achieve the maximum throughput, performance and decrease the response time. Here in this paper we will discuss the 5 main load balancing algorithms/techniques given by the researchers. Following are the Load Balancing Techniques which are in use:-

3.1 Round Robin Load Balancing Algorithm:

Round Robin algorithm is random sampling based algorithm. It means that it selects the request one by one and randomly place them to servers (nodes) irrespective of whether it is heavily loaded or lightly.

3.2 Server-based Load Balancing for Internet Distributed Services:

M. Nakai et al. [1] proposed a new server based load balancing policy for web servers which spread all over the world. It helps in reducing the service response times by using some set of rules that limits the redirection of requests to the closest remote servers without overloading them. A middleware is described to implement this protocol. It also uses a heuristic to help web servers to endure overloads.

3.3 Scheduling Strategy on Load Balancing of Virtual Machine Resources:

J. Hu et al. [2] proposed a scheduling strategy on load balancing of VM resources that uses historical data and current state of the system. This strategy achieves load balancing and reduced dynamic migration by using a genetic algorithm. It helps in resolving the issue of load-imbalance and high cost of migration thus achieving better resource utilization.

3.4 Central Load Balancing Policy for Virtual Machines:

A.Bhadani et al. [3] proposed a Central Load Balancing Policy for Virtual Machines (CLBVM) that balances the load evenly in a distributed virtual machine/cloud computing environment. This Load Balancing technique improves the overall performance of the system but does not consider the systems that are fault-tolerant.

3.5 A Task Scheduling Algorithm Based on Load Balancing:

Y. Fang et al. [4] discussed a two-level task scheduling mechanism based on load balancing to meet dynamic requirements of users and obtain high resource utilization. It achieves load balancing by first mapping tasks to virtual machines and then virtual machines to host resources thereby improving the resource utilization and overall performance of the cloud computing environment but it does not improves the response to request ratio.

4. PROPOSED WORK

4.1 Enhanced Equally Distributed Load Balancing Algorithm

Enhanced equally distributed load balancing algorithm handles the requests with priorities. It is a distributed algorithm by which the load can be distributed not only in a balanced manner but also it allocates the load systematically by checking the counter variable of each data center. After checking, it transfers the load accordingly i.e. the minimum value of the counter variable will be chosen and the request is handled easily and takes less time, and gives maximum throughput. It is a distributed technique in which the load balancer allocates the load of the job in hand into multiple web servers. The randomly transfer of load can cause some server to heavily loaded while other server is lightly loaded. If the load is equally distributed it not only improves performance also reduces the time delay. So the analysis on the load distribution algorithms the efficient scheduling and resource allocation is a critical task in case of cloud computing and also to improve the response time and processing time. While considering the impact of cost optimization one has to think about the solution to this problem. This algorithm not only balances the load also it increases the response time for the cloud.

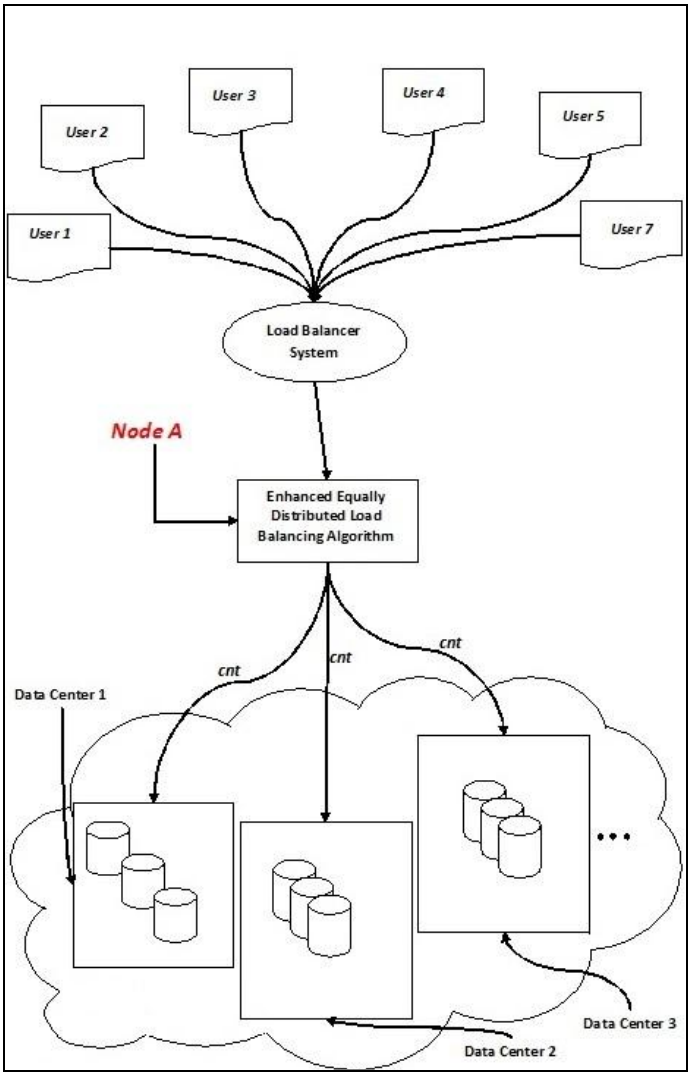

Fig -1: Enhanced Equally Distributed Load Balancing Algorithm

4.2 The Steps for efficient Distributed Load balancing through this Algorithm

This algorithm simply allocates request which is coming from the client nodes to the lightly loaded server cluster (Data Center) and gives the response in less amount of time by doing this, it makes the algorithm efficient for response to request ratio. We can see that the clients at a same time make requests to access the cloud application over the internet. Now in this algorithm all the request goes through the load balancer system by which the Node A checks the counter variable which is set to the maximum requests handled by a server cluster e.g. 300 requests per server cluster i.e. each server in the cluster can handle up to 100 request simultaneously. Let us assume that the cluster 1 is having a counter value to 250, cluster 2 is having the counter variable to 220 and cluster 3 to 270 i.e. cluster 2 is handling the smaller number of requests compared to cluster 1 and cluster 3, so here the load balancer will balances the load (requests) to cluster 2 as it is less hence the balancing is done at this level. Till now we have handled the request but how the counter variable will gets updated? The answer is the servers which the counter variable is associated with, will simultaneously changes (updates) the counter variable i.e. when a response is given back to the client the server automatically decreases its counter variable by the number 1,so that every time the algorithm will have the updated value of counter variable. Hence with the help of this algorithm the requests are handled easily by Server Clusters. The Strength of server can be increased or decreased by the service provider on request and for data centers too. So no need of Round Robin Balancing or any other technique where time is consumed and response to request ratio is low for vast number of requests. Clearly we can see over here if we assume our example that this load balancer can balances 900 requests at a time without any delay in time and responses can be given back to the clients.

5. PERFORMANCE ANALYSIS

When it comes to performance analysis of the cloud system, there are some metrics on which the analysis can be done. Some of the important metrics like Performance, Resource Utilization, Scalability, Fault Tolerance and Response Time. For a better system these are some important attributes which has to be maintained for a perfect system.

5.1 Performance:

It is used to check how efficient the system and how the performance remains steady under lot of pressure. In this algorithm the performance of whole system has increased. As we have said earlier that this system can perform many requests simultaneously i.e. by taking the help of above example 900 requests in a time so the performance of this algorithm not only stays steady but also efficient to under pressure because of the dynamic behavior of the Algorithm.

5.2 Resource Utilization:

It is used to check the utilization of the available resources given to the cloud. In this metric the algorithm works smooth i.e. on utilization of resources this algorithm uses the resources wisely and performs the task efficiently.

5.3 Scalability:

It is used to check whether the system is functional after it is scaled to any amount. Every Algorithm must comply with this metric i.e. if there is a need to expand the system to an extent the algorithm has to adjust itself to continue giving its services. This algorithm is made for such scalable requirements, because here the counter variable plays the key role when there is an increase in the number of servers or data centers this algorithm dynamically adjusts itself to that scenario and performance is maintained.

5.4 Fault Tolerance:

It is type of metric in which it is measured that how the system will perform under the node (server) failure. In this algorithm the counter variable is feeding back the requests handled by the data centers. Now, if any node fails or any node is having a fault this algorithm automatically updates the counter variable and stops itself from updating further, by doing this the algorithm itself judges that some fault occurred in that node or datacenter so the algorithm maintains its flexibility.

5.5 Response Time:

It is an amount of time taken by the system to give the response of the request given by the client. The main prospect why this algorithm is proposed is this metric. In Cloud Computing the client need the data to be accessed as fast as like he is accessing it in his home computer so response time in this algorithm is dramatically reduced from the rest of the algorithms. This can be seen by the above given examples that at a time the algorithm is giving the 900 responses back to the client which is a great figure. So the Response time for this algorithm is very good.

6. RESULTS

The results for this algorithm have been observed on the basis of the above explained scenario. We have used 3 different data centers in our cloud, and the operation is performed. For this we have applied above explained 5 algorithms and response time has been noted down. Now here on Y-axis we have Requests on servers and on X axis we have 6 different algorithms and their respective performances are shown in Bars. We can see here that each load balancing algorithm though produces good response to request ratio but Enhanced Equally Load Balancing Algorithm (EEDLBA) has the highest number responses to the requests compared to others.

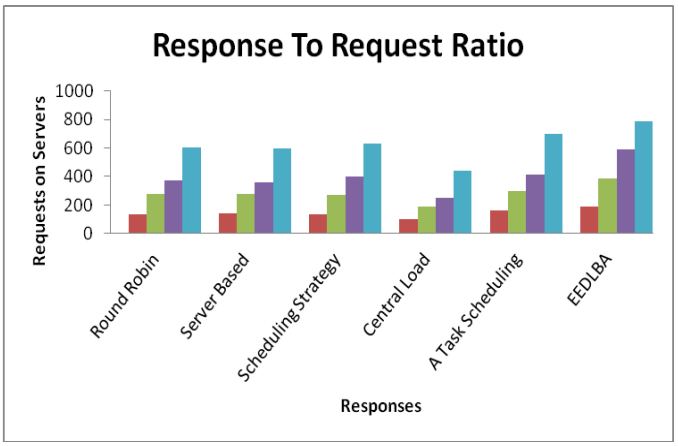

Chart -1: Response to Request Ratio Chart

Similarly, after observing the results of response time, rest four matrices are also observed and we discovered that the equally enhanced distributed load balancing algorithm is more superior to the other load balancing algorithms though there are some issues also e.g. Energy Management, Carbon Emission etc. but still the said algorithm works fine. The Performance, Resource Utilization, Scalability, Fault Tolerance and the response time is much better than the others.

CONCLUSIONS

Cloud Computing System has widely been adopted by the industry, though there are many existing issues like Load Balancing, Migration of Virtual Machines, Server unification, etc. which have not been yet fully addressed. On the Contrary the Load Balancing is the most central issue in the System i.e. to distribute the load in an efficient manner. It also ensures that every computing resource is distributed efficiently and fairly. Existing Load Balancing techniques/Algorithms that have been studied mainly focus on reducing overhead, reducing the migration time and improving performance etc., but none of them have considered the response to request ratio. The response time is a challenge of every engineer to develop the products that can increase the throughput in the cloud based sector. The several strategies lack efficient scheduling and load balancing resource allocation techniques leading to increased operational cost. This proposed algorithm not only rectifies the said issues but also reduces the request to response ratio.

REFERENCES:

[1]. A. M. Nakai, E. Madeira, and L. E. Buzato, "Load Balancing for Internet Distributed Services Using Limited Redirection Rates", 5th IEEE Latin-American Symposium on Dependable Computing (LADC), 2011, pages 156-165.

[2]. J. Hu, J. Gu, G. Sun, and T. Zhao, "A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment", Third International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), 2010, pages 89-96.

[3]. A. Bhadani, and S. Chaudhary, "Performance evaluation of web servers using central load balancing policy over virtual machines on cloud", Proceedings of the Third Annual ACM Bangalore Conference (COMPUTE), January 2010.

[4]. Y. Fang, F. Wang, and J. Ge, "A Task Scheduling Algorithm Based on Load Balancing in Cloud Computing", Web Information Systems and Mining, Lecture Notes in Computer Science, Vol. 6318, 2010, pages 271-277.

[5]. Nidhi Jain Kansal and Inderveer Chana," Cloud Load Balancing Techniques: A Step towards Green", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 1, January 2012,pages 238-246.

[6]. Nikita, Shaveta and Guarav Raj, "Comparative Analysis of Load Balancing Algorithms in Cloud Computing", International Journal of Advanced Research in Computer Engineering & Technology Volume 1, Issue 3, May 2012, pages 120-124.

[7]. Amazon Elastic Compute Cloud (EC2), http://www.amazon.com/gp/browse.html?node=201590011

[8]. Load Balancing (Computing) Wikipedia, http://en.wikipedia.org/wiki/Load_balancing_ (computing)

[9]. Load Balancing and Cloud Computing (SCVMM 2012), http://social.technet.microsoft.com/wiki/contents/articles/3937.load-balancing-and-cloud-computing-scvmm-2012.aspx

[10].Cloud Computing Patterns, Elastic Load Balancer, http://cloudcomputingpatterns.org/?page_id=269

[11]. Y. Zhao, and W. Huang, "Adaptive Distributed Load Balancing Algorithm based on Live Migration of Virtual Machines in Cloud", Proceedings of 5th IEEE International Joint Conference on INC, IMS and IDC, Seoul, Republic of Korea, August 2009, pages 170-175.

[12]. M. Randles, D. Lamb, and A. Taleb-Bendiab, "A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing", Proceedings of 24th IEEE International Conference on Advanced Information Networking and Applications Workshops, Perth, Australia, April 2010, pages 551-556.

[13]. Cloud Load Balancing, The KEMP GEO LoadMaster and the hybrid cloud a perfect cloud load balancing combination.
http://www.kemptechnologies.com/products/solutions/kemp-geo-loadmaster-and-hybrid-cloud-perfect-cloud-load-balancing-combination

[14]. K. M. Nagothu, B. Kelley, J. Prevost, and M. Jamshidi, "Ultra low energy cloud computing using adaptive load prediction", Proceedings of IEEE World Automation Congress(WAC) , Kobe, September 2010, pages 1-7.

[15]. B. P. Rima, E. Choi, and I. Lumb, "A Taxonomy and Survey of Cloud Computing Systems", Proceedings of 5th IEEE International Joint Conference on INC, IMS and IDC, Seoul, Korea, August 2009, pages 44-51.

BIOGRAPHIES:

Shreyas Mulay received his Bachelor's Degree in Computer Science and Engineering From Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, Madhya Pradesh, India in 2011.At present, he is an M.Tech. Candidate in Computer Science & Engineering Department at Amity University Rajasthan (AUR), Jaipur, Rajasthan, India His Research Interest lies in Cloud Computing.

Sanjay Jain working as an Assistant Professor in Amity University Rajasthan (AUR) .He has published many National and International Papers. His Research interests include Cloud Computing, its Security etc and are having 14 years of teaching and Research experience