Volume 4, Issue 3, March 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Enhanced Load Balancing Algorithm using Generic Gossip Protocol in a Large Cloud Environment

A.SARANYA
M.E(CSE)
Fatima Michael College of Engineering & Technology, India

S.GANESH
Assistant Professor
Fatima Michael College of Engineering & Technology, India

Abstract---Cloud computing is an emerging computing technology. In a large cloud environment the problem arises with the resource allocation. Under CPU and memory constraints we formalize the resource allocation problem. We propose a generic gossip protocol for resource allocation. Generic gossip protocol which aims at minimizing power consumption through server consolidation, while satisfying a changing load pattern. The generic Gossip protocol is otherwise called GRMP-Q, provides an efficient heuristic solution. Load balancing is one of the main challenges in cloud computing .It helps in optimal utilization of resources and hence in enhancing the performance of the system. A few existing scheduling algorithms can maintain load balancing and provide better strategies through resource allocation techniques. To gain maximum profits with optimized load balancing algorithms, it is necessary to utilize resources efficiently. This paper discusses some of the existing load balancing algorithms in cloud computing and also their challenges.

Keywords: Cloud computing, GRMP-Q, Load Balancing.

I. Introduction:

Resource Management is an important issue in cloud environment.Cloud computing is the delivery of computing and storage capacity as a service to a community of end-recipients. Cloud computing entrusts services with a user's data, software and computation over a network. Our contribution includes resource allocation which is performed by maximizing the cloud utility. The cloud utility is maximized by equally using the cloud (virtual machine) resources. Aim is to maximum utility of cloud resources and load balancing of virtual machines in the cloud. The scope of the project is to allocate cloud resources to client process. Input is the process and which it can be 0 selected by the client and proposed output is resource allocation for the given client's process in cloud. The resource allocation has done using gossip protocol. It is a set of rules with constraints for resource allocation. Constraint used in the project is physical free memory of virtual machines in the cloud and processing memory.

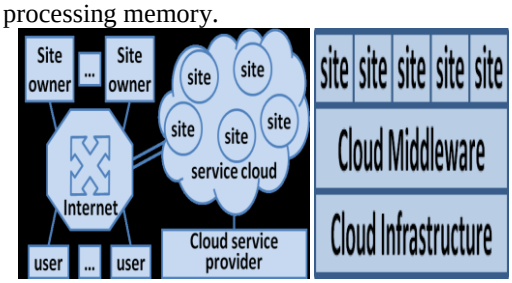


Fig. 1. (a) Deployment scenario with the stakeholders of the cloud environment considered in this work.
(b) Overall architecture of the cloud environment; this work focuses on resource management performed by the middleware layer.

In a Cloud, Cloud services are provided service provider owns and administrates the physical infrastructure.There is a number of Site owners provide services to their respective users .It is only hosted by the cloud service provider.

II. Objectives:

1. To study the performance of some of the existing load balancing algorithms
2. To design and develop the concept of load balancing using Divisible Load Scheduling Theory (DLT) for the clouds of different sizes

III. Virtualization Techniques:

Virtualisation means "something which isn't real", but gives all the facilities of a real. It is the software implementation of a computer which will execute different programs like a real machine. Virtualisation is related to cloud, because using virtualisation an end user can use different services of a cloud.

2 types of virtualization are found in case of clouds as given in [1] :

_ Full virtualization

_ Paravirtualization

3.1 Full Virtualization:
In case of full virtualisation a complete installation of one machine is done on the another machine. It will result in a virtual machine which which will have all the softwares that are present in the actual server.

3.2 Paravirtualization
In paravitualisation, the hardware allows multiple operating systems to run on single machine by efficient use of system resources such as memory and processor. e.g. VMware software. Here all the services are not fully available, rather the services are provided partially.
Paravirtualization has the following advantages :
_ Disaster recovery
_ Migration
_ Capacity management

IV. Load Balancing:

It is a process of reassigning the total load to the individual nodes of the collective system to make resource utilization effective and to improve the response time of the job, simultaneously removing a condition in which some of the nodes are over loaded while some others are under loaded. A load balancing algorithm which is dynamic and it does not consider the previous state or behavior of the system, that is, it depends on the present behavior of the system. The important things to consider while developing such algorithm are : estimation of load, comparison of load, stability of different system, performance of system, interaction between the nodes, nature of work to be transferred, selecting of nodes and many other ones [4] . This load considered can be in terms of CPU load, amount of memory used, delay or Network load.

4.1 Goals of Load balancing
The goals of load balancing are :

_ To improve the performance ubstantially
_ To have a backup plan in case the system fails even partially
_ To maintain the system stability
_ To accommodate future modification in the system

4.2 Types of Load balancing algorithms
Types of Load Balancing Algorithm
_ Sender Initiated
_ Receiver Initiated

4.2.1 Dynamic Load balancing algorithm
_ Symmetric: It is the combination of both sender initiated and receiver initiated Depending on the current state of the system, load balancing algorithms can be divided into 2 catagories:
_ Static: It doesnt depend on the current state of the system.
_ Dynamic: Decisions on load balancing are based on current state of the system. No prior knowledge is needed.

V. Existing Algorithm

Step 1. Index table can be maintained by the Load Balancer of VMs and there will be a number of requests allocated to the VM. In the beginning all VM's have 0 allocations.
Step 2. From the DataCenterController arrives, a request to allocate a new VM can be determined, it parses the table and least loaded will be identified in VM. If there are more than one, the first identified is selected.
Step 3. The Load Balancer returns the VM ID to the DataCenterController.
Step 4. The DataCenterController sends the request to the VM and it can be identified by that id.
Step 5. DataCenterController notifies the Load Balancer of the new allocation and it is finalised.
Step 6. The Load Balancer updates the allocation table incrementing the allocations count for that VM.
Step 7. When the VM finishes processing the request, and the DataCenterController receives the response cloudlet, it notifies them.
Step 8. The Load Balancer updates the allocation table by decrementing the allocation count for the VM by one.
Step 9. Continue from step 2.

VI. Proposed Algorithm: Enhanced Load Balancing Algorithm using Generic Gossip protocol.

Step1. Initially VM status will be 0 as all the VMs are available. Cloud Manager in the datacenter maintains a data structure comprising of the Job ID, VM ID and VM Status.
Step2. When there is a queue of requests, the cloud manager parses the data structure for allocation to identify the least utilized VM. If availability of VMs is more then, the VM with least hop time is considered.
Step3. The Cloud Manager updates the data structure automatically after allocation.

Step4. The Cloud Manager periodically monitors the status of the VMs for the distribution of the load, if an overloaded VM is found, and then the manager migrates the load of the overloaded VM to the underutilized VM.
Step5. The decision of selecting the underutilized VM will be based on the hop time. The VM with least hop time is considered.
Step6. Accordingly on a time to time basis the Cloud Manager updates the data structure by modifying the entries.
Step7. The cycle repeats from Step2.

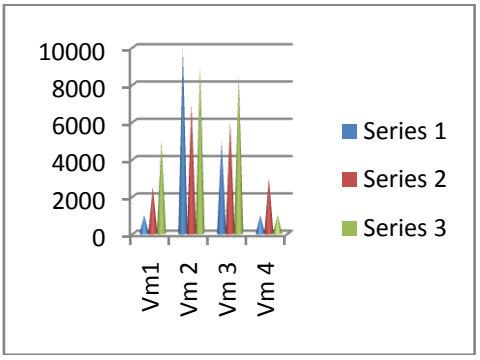
VII. SIMULATION SETUP AND EXPECTED RESULTS

User Configuration

Job ID	Job Capacity
J1	100
J2	1000
J3	10
J4	10000

Data center configuration

VM ID	VMCapacity	Series1	Series2	Series3
Vm1	1000	1000	2500	5000
Vm2	100	10000	7000	9000
Vm3	2000	5000	6000	8500
Vm4	1000	1000	3000	1000



VIII. **Conclusion**:

The load balancing is implemented in the cloud computing environment to provide on demand resources with high availability. But the existing load balancing approaches suffers from various overhead and also fails to avoid deadlocks when there more requests competing for the same resource at a time when there are resources available are insufficient to service the arrived requests. The enhanced load balancing approach using the efficient cloud management system is proposed to overcome the aforementioned limitations. The evaluation of the proposed approach will be done in terms of user configuration and data center configuration during the migration process of the load balancing approach to avoid deadlocks.

References:

[1] G. Pacifici, W. Segmuller, M. Spreitzer, and A. Tantawi, "Dynamic estimation of CPU demand of web traffic," in valuetools '06: Proceedings of the 1st international conference on Performance evaluation methodolgies and tools. New York, NY, USA: ACM, 2006, p. 26.
[2] D. Carrera, M. Steinder, I. Whalley, J. Torres, and E. Ayguade,"Utility-based placement of dynamic web applications with fairness goals," in Network Operations and Management Symposium,2008. NOMS 2008. IEEE, april 2008, pp. 9 –16.
[3] S. Voulgaris, D. Gavidia, and M. van Steen,"CYCLON:Inexpensive embership management for unstructured p2p overlays," Journal of Network and Systems Management, vol. 13, no. 2, pp. 197–217, 2005.
[4] C. Tang, M. Steinder, M. Spreitzer, and G. Pacifici, "A scalable application placement controller for enterprise data centers," in WWW '07: Proceedings of the 16th international conference on World Wide Web. New York, NY, USA: ACM, 2007, pp.331–340.
[5] M. Jelasity, A. Montresor, and O. Babaoglu, "Gossip-based aggregation in large dynamic networks," ACM Trans. Comput.Syst., vol. 23, no. 3, pp. 219–252, 2005.
[6] H. Kellerer, U. Pferschy, and D. Pisinger, Knapsack problems.Springer-Verlag, 2004.

[7] G. B. Dantzig, "Discrete-Variable Extremum Problems," OPERATIONS RESEARCH, vol. 5, no. 2, pp. 266–288, 1957.

[8] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker,"Web caching and Zipf-like distributions: evidence and implications,"in INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies.Proceedings. IEEE, vol. 1, 21-25 1999, pp. 126 –134 vol.1.

[9] HUIWen, LIN Chuang, ZHAO Hai-ying, YANG Yang, "Effective Load Balancing for Cloud-based Multimedia System", 2011 IEEE International Conference on Electronic & Mechanical Engineering and Information Technology, pp. 165-168.

[10] Wenhong Tian, Yong Zhao, Yuanliang Zhong, Minxian Xu, Chen Jing, "A Dynamic and Integrated Loadbalancing Scheduling Algorithm for Cloud Datacenters", Proc. IEEE CCIS2011, pp.311-315.

[11] Nair, T.R.Gopalakrishnan, Vaidehi, M., "Efficient resource arbitration and allocation strategies in cloud computing through virtualization", 2011 IEEE International Conference on Cloud Computing and Intelligence Systems September 2011, Beijing, China, pp. 397-401.

