Volume 5, Issue 7, July 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Load Balancing and Issue Monitoring of Cloud Nodes

Mukram B. Ansari, Prof. Patil Sachin
Department of Computer Engineering
G H Raisoni, Collge of Engineering & Management
Wagholi, Pune, India

Abstract—Cloud computing is achieving more popularity in the computation. Cloud computing is used to provide resources to client on demand and the resources may be software resources or hardware resources. Cloud computing architectures are classified as distributed or parallel. They fulfil the needs of multiple clients in different circumstances. The distributed architecture deploys resources distributive to deliver services efficiently to users in various geographical areas. In a distributed environment Clients generates request randomly in any processor and this is the major drawback which is associated with task assignment. The unequal task assignment to the processor creates imbalance. Due to this some of the processors are overloaded and some of them are under loaded. The main objective of load balancing is to distribute the load from overloaded process to under loaded process or idle nodes.

Keywords—Cloud Computing; Load Balancing; Load Balancing Algorithms; Error Rate; Throughput; Threshold

I. INTRODUCTION

A) Cloud Computing

The origin of term "cloud" is evolved from the world of telecommunications when service providers started use of virtual private network (VPN) services for data communications. National Institute of Standards and Technology (NIST) says that: "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction". Cloud computing can store the data on servers and can be accessed through internet. Due to this it's not required to store the data on desktops, portables etc. Cloud computing allows better utilization of distributed resources over a large data. The major components of cloud computing are clients, Data cenre and Servers [9]. The end user interacts with client to avail the services of cloud. The client may be mobile devices, thin clients or thick clients. The second component i.e. Data centre is collection of servers hosting multiple applications. The last and third component is distributed servers which are present throughout the Internet hosting various applications. [1][2]

B) Load Balancing

Load balancing is one of the major problem in cloud computing. The high performance, minimum response time and high resource utilization ratio can be achieved only when the different tasks between nodes in cloud network are distributed. Load balancing technique is used to distribute tasks from over loaded nodes to under loaded or idle nodes. Load Balancing is a method of sharing workload across one or more servers, network interfaces, hard drives, or other computing resources. The data centres may be subjected to risks related with any physical device, including hardware failure, power and/or network interruptions, and limited resources. It generates new opportunities and economies-of-scale to application leaders. Many of them are shifting business-centric metrics to service level management (SLM) to make IT closer and efficient to business. For better resource utilization the load in the cloud system must be balanced [11] evenly.

II. LITERATURE SURVEY

Different algorithm has been proposed by many researchers and has been discussed in many literatures. Following load balancing techniques are currently used in clouds.

In [1], author Daniel discussed the opportunities and challenges for efficient parallel data processing in clouds. Nephele is the first data processing framework to explicitly exploit the dynamic resource allocation offered by today's IaaS clouds for both, task scheduling and execution. Particular tasks of a processing job can be assigned to different types of virtual machines which are automatically instantiated and terminated during the job execution. The performance evaluation gives a first the possibility to automatically allocate/deallocate virtual machines during its a job execution.

In [2] Yi Lua, Qiaomin Xie proposed algorithms called Join-Idle-Queue (JIQ) for distributed load balancing in large systems limit. This algorithm gives better results in terms of a reduced system load. It produces 30-fold decrease in queuing overhead as compared to Power-of-Two at medium to high load.

In [3] Yunhua Deng et. al., presented solution to scalability as it becomes one of the major challenges in designing an interactive DVE system. To address this problem, two methods are presented as: uniform adjustment scheme and adaptive adjustment scheme. The first method performs a even distribution of the load among the neighbour servers. This scheme is very simple to implement. The second scheme provides limited degree of user tracking but without the need to communicate with neighbour servers. It is a better approximation method.

In [4] Tin-Yu Wu, presented data centre management architecture: Index Name Server (INS) and it is used to avoid the system burden caused by duplicate data. It integrates duplication and access point selection optimization method to improve the performance of system. Load balancing of the system can be achieved in the most efficient way when the cross-domain cost and transfer time is reduced efficiently.

In [5] Shamsollah Ghanbaria, proposed a new priority based job scheduling algorithm (PJSC) which is based on multiple criteria decision making model. Priority is an important issue of job scheduling in cloud environments. Also the author has provided a discussion about issues related such as complexity, consistency and finish time.

In [6] M.E. Frîncu, presented the new applications which is oriented towards Web 2.0. Their availability varies as per the number of resource failures and on the deviation in user hit rate. Scaling is used to solve these problems. It becomes unavailable if one or more nodes fail. The parameters like node load, closeness to the optimal solution and success rate are sed for the testing.

In [7]Chun-Cheng Lin, presented a centralized hierarchical cloud-based multimedia system. It consists of a resource manager, cluster heads, and server clusters. The resource manager assigns clients' requests for multimedia service tasks to server clusters as per the task features. Then each cluster head distributes the assigned task to the servers within its server cluster.

In [8] L.D. Dhinesh Babua, presented algorithm named honey bee behaviour inspired load balancing (HBB-LB). It designed to achieve well balanced load across virtual machines for improving the throughput. The tasks removed from these VMs are treated as honey bees, which are the information updaters globally. Honey bee behavior inspired load balancing improves the overall throughput of processing and priority based balancing focuses on reducing the amount of time a task has to wait on a queue of the VM. Thus, it reduces the response of time of VMs.

In [9] Giuseppe Aceto, presented detailed analysis of monitoring for the Cloud. The number of Cloud-based services has increased rapidly. Due to this the complexity of the infrastructures behind these services increased. For better operation and management efficient monitoring is constantly required.

In [10] Jianying Luo, presents an important energy management problem. The proposed algorithm is used exploit the temporal diversity of electricity price and dynamically schedule workload to run on IDC servers.

In [11] Jun Wang, proposed the parallel computing frameworks such as MapReduce and Hadoop to run data intensive applications and conduct analysis. Existing data parallel frameworks distribute the data using a random placement method for ease of use and load balance. The overall MapReduce job response time is reduced by 36.4%.

III. SYSTEM ARCHITECTURE

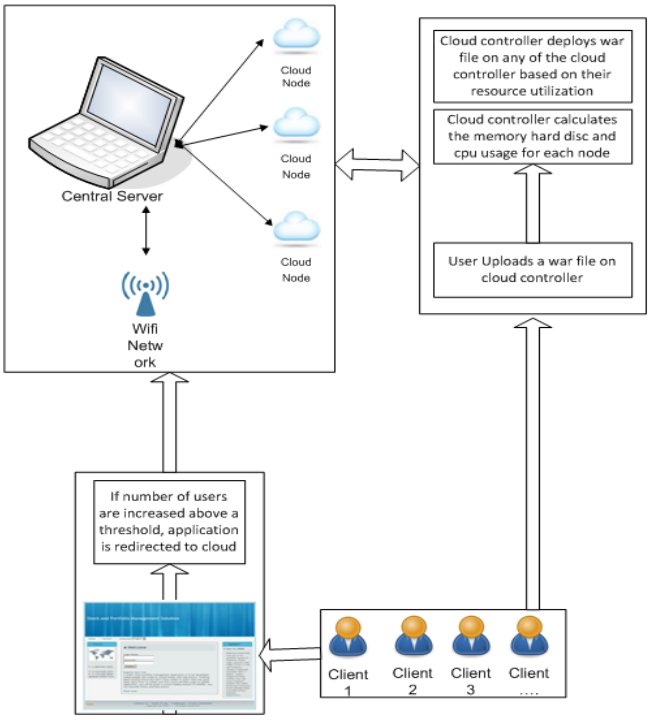

Fig. 1 Addressing scalability issue and resource monitoring in cloud nodes

Proposed Load Balancing in Cloud Computing contains User, Server, Load Balancer, and Stock Application.

Usually the Cloud computing system is divided into two parts as front end and the back end. Front end is used through which user can interact with the server. The backend is nothing but the server which provides data to the client. Between server and client network is used as middleware. To develop load balancing algorithm for cloud computing environment to distribute the load equally on servers to improve response time and processing time by transferring load from heavily loaded server to lightly loaded server. The overall aim of the proposed work is to develop a scalable Cloud solution which is capable of response to needs of Stock Broking firm with highest performance, scalability and low cost.

A) User: End users interact with the servers to manage information related to the cloud. Users are assigns task to the servers on which stock application is running.

B) Server: On requesting of user the distributed server will send the request to load balancer to check whether any node is available or not.After getting response from load balancer server will migrate task coming from user to node on which stock application is running.

C) Load Balancer: The load balancer monitors all nodes in cloud environment on which stock application is running. It calculates free RAM, free CPU and response time of each node. Then it selects one node who's RAM and CPU is less utilized and response time is very low, and sends migration link to server.

D) Stock Application: After selecting proper node for execution of user's assigned task that node will send response to user's query. The measuring application success as "user interface" alone is not sufficient in today's competitive environment.

E) Product Features

In our proposed model we establish cloud setup between two computers using Cloudbees on peer to peer network. This can be discussed as follows-

1) Cloud Setup - Creating cloud (test bed) by using (Cloudbees)

Cloud monitoring allows us to track the performance of the cloud services we might be using. Whether we are using popular cloud services such as Google App Engine, Amazon Web Services, or a customized solution, cloud monitoring ensures that all systems are going. Cloud monitoring allows us to follow response times, service availability and more of cloud

2) Resource Monitoring

monitoring critical resources like RAM, CPU, memory, bandwidth, partition information, running process information and utilization and swap usages etc. In this section we are developing an application in java where we are monitoring the node resources like RAM, CPU, Memory, Bandwidth, Partition information, Running process information and utilization.

3) Load Balancing

Load balancing algorithm for homogeneous and heterogeneous architectures. Load Balancing is a technique in which the workload on the resources of a node is shifts to respective resources on the other node in a network without disturbing the running task. Depending on the user session and the load on each web server, the load balancer forwards packets to different web servers for processing.

4) Testing

In order to evaluate the performance of complete setup, need to deploy resource monitoring and load balancing tools on test bed and evaluate performance of our algorithm.

IV. PROPOSED DYNAMIC LOAD ALGORITHM (DLA)

The DLA algorithm is used in the proposed work. In proposed method the dynamic cloud computing environment is used, the intermediate node is used to monitor the load of each VM in the cloud pool. In this approach the user can send the request to the intermediate node. It is responsible for transfer the client request to the cloud. Here, the load is considered as in terms of CPU load with the amount of memory used, delay or Network load.

The algorithm uses the six phases for load balancing as under

1) Get Load Status of All the Nodes
2) Evaluate the Status Of nodes
3) Predict The Future Load Flow
4) Benefit Estimates
5) Choose Receiver Nodes
6) Migration

V. PERFORMANCE MEASUREMENT MATRICES

Metrics are a specific calculated measurement. They help us gauge success and failure, as well as enable comparisons. Load testing tools measure key performance statistics and present those metrics to you in graphs and reports.

Focus on the key metrics because reducing data clutter can make it easier to spot bottlenecks and inefficiencies in a web application's design and implementation. Below are commonly used performance metrics for load testing:

1) *Average Response Time* – The average time it takes for a server to respond to requests. Usually it is measured in milliseconds. It shows overall performance health from the user perspective.
2) *Peak Response Time* – The longest response out of all responses for a given time period. It shows outliers, the slowest transactions, the problem areas to investigate.
3) *Error Rate* – Percentage of failures. Shows what ratio of requests is getting good versus bad responses.
4) *Throughput* – The data transferred between the user and site. It usually measured in kilobytes per second.
5) *Requests per Second* – The number of simultaneous hits on the site at approximately the same time.
6) *Concurrent Users* – The number of users holding a site session, but not necessarily sending requests simultaneously.
7) *CPU/Memory Utilization* – The percentage of CPU or memory that a web server is using at a point in time. Shows when the system is resource constrained as a root cause of performance failure.

VI. RESULTS & DISCUSSION

In the proposed work there is need of load testing tool to measure performance of user's request. This work is based on cloud environment. There are many tools available online to measure load on cloud nodes. We are using Load Storm cloud load testing tool. Load testing is like a storm of users all hitting your site at the same time and measuring the response. In broadest terms, load testing is the simulation of large amounts of usage for a particular system. The purpose of a load test is to measure the site's performance while handling simulated user activity. The test produces metrics that show the speed and scalability of a site.

1) The load test finished with scheduled to run for 20 minutes with a linear pattern, starting at 5 vusers and increasing to 10 vusers. Table 1 shows summary of result for various performance metrics used.

TABLE 1: SUMMARY OF THE RESULT

	Requests	Response (average s)	Response (max s)	RPS (average)	Throughput (average)	Total Transfer
HTML	288	0.6	1.01	0.24	23.05 kB/s	0.3 GB
Other *	1531	0.22	0.72	1.28	10.31 kB/s	0.1 GB
Total	1819	0.28	1.01	1.52	33.36 kB/s	0.4 GB

*Other includes javascript, css, images, pdf, task migration etc. (any content type except html and xml)

2) Comparative Analysis for Overall response time

Some of the load balancing techniques are tested and monitored on cloud analyst. Here Round Robin, Equally Spread Current Execution Load, and Throttled algorithms are used. These algorithms are tested on cloud analyst simulation tool and the result is given below table. Table 2 shows the comparison of overall response time for different algorithms & Figure 2 shows the response time in terms of response time.

TABLE 2: COMPARISON TABLE FOR RESPONSE TIME

Sr. No.	**Name of Algorithm**	**Overall Response Time in ms**		
		Avg	**Min**	**Max**
1	DLA	280	33.36	609.39
2	Round Robin	292.79	39.33	607.82
3	Equally Spread Current Execution Load	292.84	37.83	608.77
4	Throttled	292.79	38.51	597.84

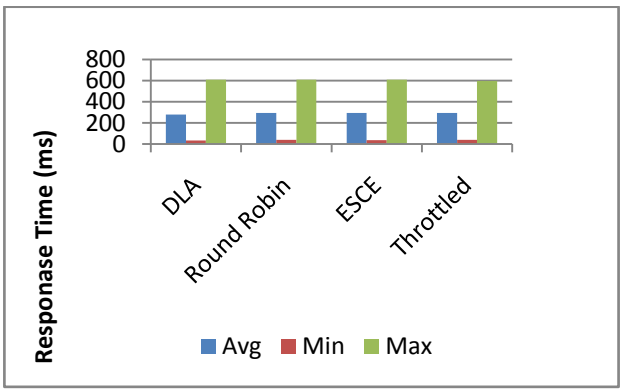

Figure 2: Comparison Chart Response Time

3) Total data transferred in between servers and virtual users are given below. The data transfer is given in Giga Byte. Table 3 shows the result for the comparative analysis for the data transfer.

TABLE 3: COMPARISON TABLE FOR DATA TRANSFER

Virtual Users	**ESCEL**	**RR**	**Throttled**	**DLA**
5	0.31	0.32	0.32	0.3
10	0.5	0.48	0.51	0.55
15	1.38	1.37	1.38	1.45
20	1.98	1.98	1.96	2

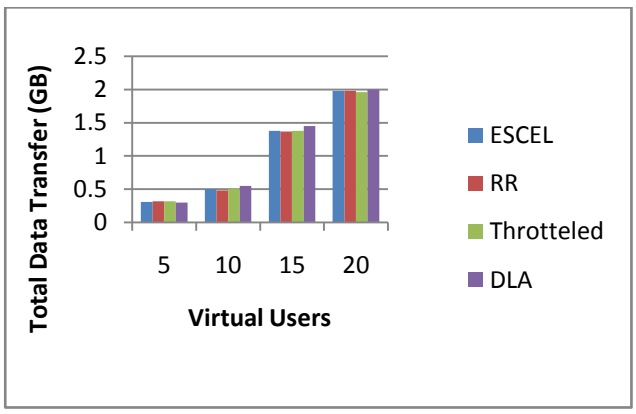

Figure 3: Comparison Chart Data Transfer

4) Table 4 indicates the load statistics on node with various User loads.

TABLE 4: COMPARISON TABLE FOR LOAD STATISTICS ON NODE

Load Statistics on Node				
	Userload	**% RAM Free**	**% CPU Free**	**Region Wise Server Distribution**
Node1	50	20	45	R1
Node2	10	10	45	R2
Node3	20	50	45	R3
Node4	20	30	45	R4

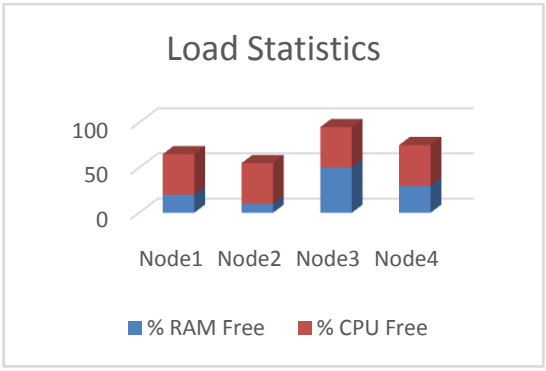

Figure 4: Comparison Chart for Load statistics on Node

5) Table 5 shows the result for Userload & additional load with fixed threshold of 50.

Table 5: Summary of the result for existing system

Existing System				
	Userload	**Additional UserLoad**	**Fixed Threshold**	**Next Node Load Balancing**
Node1	50	10	50	Node2
Node2	10	50	50	Node3
Node3	20	10	50	Node4
Node4	20	10	50	Waits

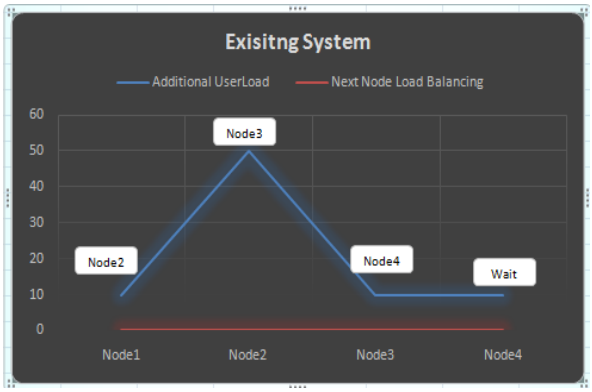

Figure 5: Comparison Chart for result for existing system

6) Table 6 shows the summary of result for Userload & additional load with fixed threshold. Here threshold values are dynamically updated and inactive user count is decreased. Node 3 is selected because it has maximum free RAM+CPU Usage. Thus shifting of load is done at 2 levels i.e. 1. Resource Level and 2 .User Level

TABLE 6: SUMMARY OF THE RESULT

Proposed Method						
	User load	**Additional UserLoad**	**Fixed Threshold**	**Next Node Load Balancing**	**% RAM Free**	**% CPU Free**
Node1	50	10	Threshold Values are dynamically updated and Inactive User count is decreased	Node3	20	45
Node2	10	50		Node3	10	45
Node3	20	10		Node3	50	45
Node4	20	10		Node3	30	45

Figure 6: Results of proposed algorithm

VII. CONCLUSION

Cloud Computing has extensively been used by the IT industry even though there are many existing issues like Load Balancing, Virtual Machine Migration, Server Consolidation, Energy Management, etc. These issues are yet have not been fully addressed. Load balancing is a central issue in which its required to distribute the excess dynamic workload evenly to all the nodes in the Cloud so that a high user satisfaction and resource utilization ratio can be achieved. Existing Load Balancing techniques mainly concentrate on reducing overhead, service response time and improving performance etc., but very few techniques has considered the execution time of any task at the run time. Therefore, there is a need to develop such load balancing technique that can improve the performance of cloud computing along with efficient resource utilization.

REFERENCES

[1] Daniel Warneke "Exploiting Dynamic Resource Allocation for Efficient Parallel Data Processing in the Cloud", IEEE Transactions On Parallel And Distributed Systems, VOL. 22, NO. 6, JUNE 2011

[2] Yi Lua, Qiaomin Xie , "Join-Idle-Queue: A novel load balancing algorithm for dynamically scalable web services", SciVerse Science Direct , Performance Evaluation 68 (2011) 1056–1071, Published by Elsevier B.V. doi:10.1016/j.peva.2011.07.015

[3] Yunhua Deng and Rynson W.H. Lau , "On Delay Adjustment for Dynamic Load Balancing in Distributed Virtual Environments", IEEE Transactions On Visualization And Computer Graphics, VOL. 18, NO. 4, APRIL 2012, 1077-2626© 2012 IEEE Published by the IEEE Computer Society

[4] Tin-Yu Wu, Wei-Tsong Lee, "Dynamic Load Balancing Mechanism based on Cloud Storage", 978-1-4577-1719-2/12/$26.00 ©2012 IEEE

[5] Shamsollah Ghanbaria "A Priority based Job Scheduling Algorithm in Cloud Computing", International Conference on Advances Science and Contemporary Engineering 2012 (ICASCE 2012), Procedia Engineering 50 (2012) 778 – 785, 1877-7058 © 2012 Elsevier B.V, doi: 10.1016/j.proeng.2012.10.086

[6] M.E. Frîncu , "Scheduling highly available applications on cloud environments", SciVerse ScienceDirect, Future Generation Computer Systems, 0167-739X © 2012 Elsevier B.V. ,doi:10.1016/j.

[7] Chun-Cheng Lin, Hui-Hsin Chin, "Dynamic Multiservice Load Balancing in Cloud-Based Multimedia System", IEEE SYSTEMS JOURNAL 1932-8184/$31.00 _c 2013 IEEE

[8] L.D. Dhinesh Babua, , P. Venkata Krishnab, "Honey bee behavior inspired load balancing of tasks in cloud computing environments", SciVerse ScienceDirect Applied Soft Computing, 1568-4946/$ – see front matter © 2013 Elsevier B.V. All rights reserved. http://dx.doi.org/10.1016/j.asoc.2013.01.025

[9] Giuseppe Aceto, Alessio Botta, "Cloud monitoring: A survey", SciVerse ScienceDirect Computer Networks, 1389-1286- 2013 Elsevier B.V. All rights reserved. http://dx.doi.org/10.1016/j.comnet.2013.04.001

[10] Jianying Luo, Lei Rao, and Xue Liu , "Temporal Load Balancing with Service Delay Guarantee for Energy Cost Optimization in Internet Data Centers", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, Digital Object Indentifier 10.1109/TPDS.2013.69 1045-9219/13/$31.00 © 2013 IEEE

[11] Jun Wang, Qiangju Xiao , "DRAW: A New Data-gRouping-AWare Data Placement Scheme for Data Intensive Applications With Interest Locality", IEEE TRANSACTIONS ON MAGNETICS, VOL. 49, NO. 6, JUNE 2013, Digital Object Identifier 10.1109/TMAG.2013.2251613

