e-ISSN: 2349-9745
p-ISSN: 2393-8161
Scientific Journal Impact Factor (SJIF): 1.711
International Journal of Modern Trends in Engineering and Research
www.ijmter.com

A Model for load balancing for the Public Cloud by cloud partitioning technique

Priyanka Shinde[1], Prof. P.M.Chawan[2]
[12]Computer Engineering and Information Technology / VJTI,Mumbai

Abstract— Cloud Computing is growing rapidly and clients demand more services and better results, so load balancing for Cloud is important research area. Currently , the usage of internet and related resources has increased tremendously. Because of this there is huge increase in workload which causes uneven distribution of workload that results in server overloading and may crash. In such systems the resources are not optimally used. This degrades the and efficiency. Good load balancing makes cloud computing more efficient and improves user satisfaction. This article introduces a better load balance model for the public cloud based on the cloud partitioning concept with a switch mechanism to choose different strategies in different situations. Model contains main controller that chooses the suitable partitions for arriving jobs while the balancer for each cloud partition chooses the best load balancing strategy.

Keywords— cloud computing , cloud partition, Load balancing , Load balancing algorithm, Public Cloud, switch mechanism.

1. INTRODUCTION

CLOUD computing is a parallel and distributed system which helps on demand network access to shared pool of computing resources like applications, storage and servers which can be accessed by the user on pay per basis.

Load balancing distributes workloads across multiple computing resource which ultimately results into optimize resource use, minimize response time, maximize throughput, and avoid overload of any single resource. Load balancing in cloud computing systems has become really a challenge now. This requires continues innovative solutions in cloud environment. It is not always practically cost efficient or feasible to maintain one or more idle services such as storage, servers and applications just as to fulfill the required demands. It is important to control workloads to improve system performance and maintain stability.

The load on every cloud is flexible and dependent on various factors. To handle this problem of imbalance and to increase its work efficiency, this paper tries to implement "A Model for load balancing for the Public Cloud by cloud partitioning technique". Load balancing algorithms are classified as static and dynamic algorithms [6].

Static algorithms are mostly appropriate for homogeneous and steady environments and can produce better results for the same. However, they are usually not elastic and cannot match the dynamic changes to the attributes during the run time. Dynamic algorithms are more elastic and take into consideration different types of attributes in the system both prior to and during execution-time. These algorithms can adapt to changes and provide better results in heterogeneous and dynamic environments.

However, as the distribution attributes become more complex and dynamic. Model described in this paper has a main controller, balancers, servers and clients which introduces a switch mechanism to choose different strategies for different situations. This model divides the public cloud into cloud partitions and applies different strategies to balance the load on cloud. The basic designs of the system and algorithms to implement it are described in this paper [1].

@IJMTER-2015, All rights Reserved

2. CURRENT ISSUE AND CHALLENGES

Cloud computing is effective and scalable but to maintain the stability of processing many jobs in the cloud computing environment is a complex problem. The arrival pattern of jobs is not predictable and the capacities of each node in the cloud differ also controlling the workload is difficult to improve performance.

2.1 Challenge In Proposed Model

Before examining the current load balancing approaches for Cloud Computing, we need to identify the challenges involved that could affect the performance of the algorithm proposed in this paper. We will discuss the issues to be addressed while attempting to present an optimal solution to the issue of load balancing in Cloud Computing. These challenges are summarized in the following points.

2.2 Cloud partition in Spatial Distribution of the Cloud Nodes

It is a challenge to design a load balancing algorithm that will work for spatially distributed nodes because of the factors such as the distance between the client, the speed of the network links among the nodes and the distances between the nodes involved in providing the service. There is a need to develop a way to control load balancing mechanism among nodes which will effectively tolerate high delays [3].

2.3 Evaluation of load status for any node in any Partition

A good algorithm is needed to set High Load degree and Low Load degree, and the evaluation mechanism needs to be more comprehensive.

2.4 To find load balancing strategy

Many tests are needed to guarantee system availability and efficiency and to compare different strategies.

3. RELATED WORK

Many studies have been done for load balancing of the cloud environment. Load balancing in cloud was briefed in a white paper written by Adler[2] who introduced the tools and techniques used for load balancing in the cloud. However, load balancing in cloud is still a problem that needs new architectures to adapt too many changes. Nishant et al.[4] used the ant colony optimization methods in nodes load balancing. Randles et al.[5] gave a compared analysis of some of algorithms in cloud computing by checking the performance time and cost.

Chaczko et al.[3] described the role that load balancing has in improving the performance and maintaining stability. There are many load balancing algorithms, such as Round Robin, Game theory Algorithm, and Ant Colony algorithm.

We will use the Round Robin algorithm here because it is fairly simple.

4. MODEL FOR LOAD BALANCING

Load balancing is needed to distribute the dynamic workload evenly across all the nodes which helps to achieve high resource utilization ratio and user satisfaction by ensuring optimize allocation of all computing resource.

Load balancing based on Cloud Partitioning of public cloud

A public cloud is based on the standard cloud computing model which includes many nodes in different geographical locations. Cloud partitioning is used to manage this large public cloud. This large public cloud is divided into many subareas based on the geographic locations. The architecture for the same is shown in Fig.1. Once cloud partitions are created. We can start with the load balancing, when a job arrives at the system, the main controller assigns jobs to suitable cloud partition and then communicates with the balancers in each partition to refresh this status information. When the load status of a cloud partition =

@IJMTER-2015, All rights Reserved

normal, job assignment is done locally. But if it is not normal, then this job is transferred to other partition. Now we will discuss some load balancing technique for both the partition having either load status as idle or load status as normal based on load degree. The load degree of the node is based on different static and dynamic parameters of each node.

Role of Main Controller and balancers

Main controller and balancers has the job of load balancing. The main controller assigns jobs to suitable cloud partition and then communicates with the balancers in each partition to refresh this status information. The balancers in each partition collect the status information from every node and then choose the right strategy to distribute the jobs.

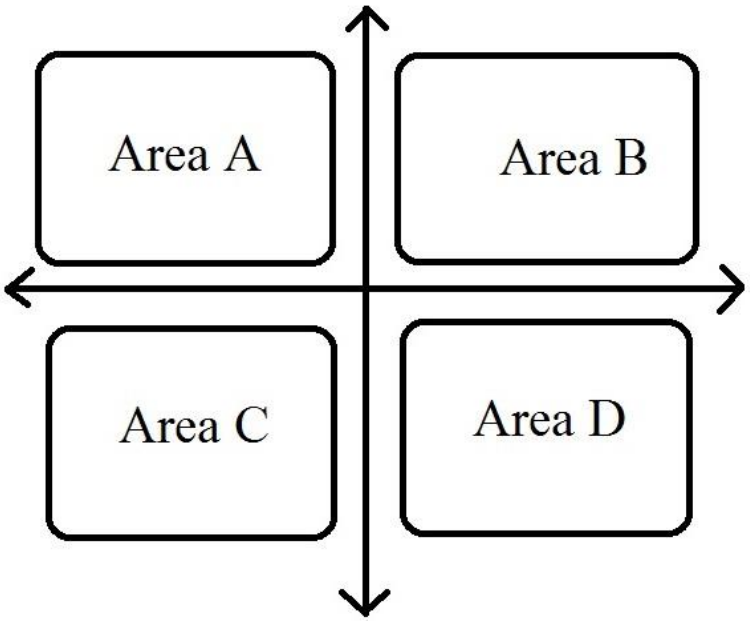

Fig 1. Load balancing architecture

Job assignment to cloud partition

When a job arrives at the public cloud, the main controller need to choose the right partition. The status of the cloud partition is divided into three types:

(1) Idle: When the percentage of idle nodes exceeds threshold, change to idle status.

(2) Normal: When the percentage of the normal nodes exceeds threshold, change to normal load status.

(3) Overload: When the percentage of the overloaded nodes exceeds threshold, change to overloaded status.

The thresholds are set by the cloud partition balancers. The main controller has to communicate with the balancers frequently to refresh the status information and then it dispatches the jobs.

We use the following algorithm for assigning job to nodes.

BEST PARTITION SEARCHING ALGORITHM

```
Begin
While job do
searchBestPartition(job);
if partitionstate==idle || partitionState  == normal
```

@IJMTER-2015, All rights Reserved

then
Send job to Partition ;
else
search for another partition;
end if
end while
end.

Assigning jobs to the nodes after choosing cloud Partition :
Cloud partition collects all status information from the node and computes load degree of the partition. Various static and dynamic parameters affects load degree of the partition. Static parameters are memory speed of CPU, number of CPU include in load balancing and memory size. Dynamic parameters is CPU utilization ratio, Network bandwidth, memory utilization ratio[1].

Process of Calculating Load degree for each node

1. Define a load parameter set:
F= {F1,F2…Fm} , m presents the total number of the parameters.
2. Compute the load degree as Load Degree(N)= Σ αiFi where i= 1…m , α= weights according to job , N= current node.
3. Average cloud partition degree from the node load degree statistics as: Load degree_avg= (Σi=1…n LoadDegree(Ni)) /n.
4. Three level node status are defined.

Idle : Load_degree(N)=0
Normal : 0<=Load_degree(N)<= Load_degree_high
Overloaded : Load_degree_high<=Load_degree(N)

Node is not available and can not receive jobs until it returns to the normal status. Each balancer maintains Load Status Table. The load degree results are input into the Load Status Tables. Each balancer refreshes Load Status Table in fixed period T which is then used by the balancers to calculate the partition status. Each partition status has a different load balancing solution and accordingly Load Balancing strategy is applied

5. CLOUD PARTITION LOAD BALANCING STRATEGY

For Idle Status
Cloud partition may process the jobs quickly when the cloud partition is idle, thus enabling a simple load balancing method. In Round Robin algorithm every node has an equal opportunity. It may be used here for simplicity. Public cloud may contain many nodes which are geographically dispersed and configuration and the performance of each node also vary; due to this method may overload some of the nodes.
Thus “Round Robin based on the load degree evaluation” which is an improved Round Robin algorithm can be used. This algorithm is very much similar to simple Round Robin algorithm with minor changes. Before assigning jobs in Round Robin manner, the nodes in the load balancing table are ordered based on the load degree from the lowest to the highest. The system then builds a circular queue and walks through this queue again and again. Jobs will then be assigned to nodes according to the load balancing table. The node order will be changed when the balancer refreshes the Load Status Table.

@IJMTER-2015, All rights Reserved

For Normal Status

Jobs are arriving much faster than in the idle state and the situation is far more complex when the cloud partition is normal. Each user wants his jobs completed in the shortest time, so the public cloud needs a method that can complete the jobs of all users with reasonable response time.

Penmatsa and Chronopoulos[13] proposed a static load balancing strategy based on game theory for distributed systems. Aote and Kharat[15] gave a dynamic load balancing model based on game theory. These two strategies can be used for load balancing in normal status.

6. ACKNOWLEDGEMENTS

I would like to thank all those people whose support and co-operation has been a valuable asset during the course of this Paper. I would also like to thank our Guide **Prof. P. M. Chawan** for guiding me for completing this paper.

REFERENCES

[1] Gaochao Xu, Junjie Pang, and Xiaodong Fu* "A Load Balancing Model Based on Cloud Partitioning for the Public Cloud ", ISSN l1007-0214l l04/12l lpp34-39,Volume 18, Number 1, February 2013.

[2] B. Adler, Load balancing in the cloud: Tools, tips and techniques, http://www.rightscale. com/info center/whitepapers/ Load-Balancing-in-the-Cloud.pdf, 2012. Moore, Interval analysis (Englewood Cliffs, NJ: Prentice-Hall, 1966).

[3] Z. Chaczko, Venkatesh Mahadevan, Shahrzad Aslanzadeh, Christopher Mcdermid. "Availability and Load Balancing in Cloud Computing" 2011 International Conference on Computer and Software Modeling.

[4] K. Nishant, P. Sharma, V. Krishna, C. Gupta, K. P. Singh,N. Nitin, and R. Rastogi, Load balancing of nodes in cloud using ant colony optimization, in Proc. 14th International Conference on Computer Modeling and Simulation (UKSim), Cambridgeshire, United Kingdom, Mar. 2012, pp. 28-30.

[5] M. Randles, D. Lamb, and A. Taleb-Bendiab, A comparative study into distributed load balancing algorithms for cloud computing, in Proc. IEEE 24th International Conference on Advanced Information Networking and Applications, Perth, Australia, 2010.

[6] N. G. Shivaratri, P. Krueger, and M. Singhal, Load distributing for locally distributed systems, Computer, vol. 25, no. 12, pp. 33-44, Dec. 1992.

[7] S. Penmatsa and A. T. Chronopoulos, Game-theoretic static load balancing for distributed systems, Journal of Parallel and Distributed Computing, vol. 71, no. 4, pp. 537-555, Apr. 2011.

[8] S. Aote and M. U. Kharat, A game-theoretic model for dynamic load balancing in distributed systems, in Proc. The International Conference on Advances in Computing, Communication and Control (ICAC3 '09), New York, USA, 2009, pp. 235-238.

@IJMTER-2015, All rights Reserved