International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

Available at http://inpressco.com/category/ijcet

Research Article

Load Distribution and Balancing over Cloud using Cloud Partitioning

Snehal D. Sonawane[Å*] and R. H.Borhade[Å]

[Å]Department of Information Technology,University of Pune, State Maharastra, Country India

Accepted 10 May 2014, Available online 01 June 2014, **Vol.4, No.3 (June 2014)**

Abstract

Cloud computing technology is bringing attractive changes in IT field. One of the distributed computing paradigm which mainly focuses on providing everything as a service to the consumer is cloud computing and it provides computational or storage resources and database to users. In the cloud computing environment, load balancing has an important impact on the performance. Effective implementation of load balancing can make cloud computing more effective and it also improves user satisfaction. A better load balance model can be implemented for the large cloud which uses the cloud partitioning concept. Switch mechanism can also be used to choose different strategies for different situations. The algorithm applies the token generation algorithm to improve the effect of load balancing strategy in the public cloud environment.

Keywords: *Cloud partitioning, Load balancing model, Switch mechanism*.

1. Introduction

In case of cloud computing, the data is mainly stored as distributed manner. It is saved on remote location or virtual locations randomly. If it is going to upload the data randomly on cloud it results into an imbalance of data in cloud server storage. For example, some of the nodes are heavily loaded while other nodes are having negligible load or doing very less work i.e. one server has been loaded by 10 GB data and the second server has zero data uploaded on it.

Cloud computing can utilize the services regardless of the installation and maintenance problems to an organization. It is a technology that aims to provide on-demand services on scalable basis over the Internet via Cloud vendors to organizations.

The data that will be stored in the cloud will be able to access only by authorized users, which are specified by the cloud service provider and for this purpose access control system is used in cloud.

A large public cloud will include many nodes from different geographical locations. Cloud partitioning concept is highly used to manage this large cloud. Based on the geographic locations there is a subarea of the public cloud with divisions.

After creating the cloud partitions, the main controller starts to decide which cloud partition should receive the job. The partition load balancer decides how the job can be assigned to the nodes. This partitioning can be accomplished locally when the load status of a cloud partition is normal.

There are many researches of load balancing for the cloud technology. Load balancing technique in cloud computing was described in a white paper written by Adler. He has introduced the different tools and techniques which are commonly used for load balancing in the cloud computing. As well as, load balancing in the cloud is still a new difficulty which needs another new architecture to adapt various changes. The role that load-balancing plays in improving the performance and maintaining stability.

2. Related Works

There are many studies of load balancing technology for the cloud computing environment. There is various load balancing algorithms. One of them is Round Robin, as well as Equally Spread Current Execution Algorithm, and Ant Colony algorithm. Nishant et al are some more efficient algorithms. They used the ant colony optimization method in nodes load balancing. Randles et al. gave a comparison of analysis of some algorithms in cloud computing technology by checking the cost and the performance time. The conclusion is that the ESCE algorithm and throttled algorithm gave better results than the Round Robin algorithm. Some of the traditional loads balancing methods are very much similar to the allocation method which is used in the operating system, for example, the Round Robin algorithm and the First Come First Served (FCFS) algorithm. The only reason to use Round Robin algorithm is the simplicity of it.

There is an important role of cost/performance ratio in networks of workstations and it has been noticed that it is constantly improving. It is expected that this trend will be continued in the near future. The aggregate peak rate of the fastest parallel computers matches or exceeds the peak rate offered by such computers. Therefore, distributed computing systems are a viable and it is also a less

*Corresponding author: **Snehal D. Sonawane**

expensive alternative to parallel computers. However, in concurrent programming of a distributed system, a serious difficulty is how to deal with scheduling and load balancing of such a system which might be consist of heterogeneous computers.

The anticipated uptake of Cloud computing technology, built on well-established research work in utility computing, distributed computing, networks, Web Services and virtualization, will bring many advantages in flexibility, cost and availability for different service users. These benefits are expected to drive the demand for Cloud services further, increasing both the Cloud's customer base and the scale of various Cloud installations. There are implications for number of technical issues in various Service Oriented Architectures and Internet of Services (IoS)-type applications; including scalability, fault tolerance and high availability. The establishment of effective load balancing techniques is central to all these issues. Requiring an effective distributed solution, it is clear that the complexity and scale of these systems makes centralized assignment of jobs to particular servers infeasible.

The advanced approach of ant colony optimization is applied from the perspective of grid network systems or cloud with the main aim of load balancing of nodes. There is an edge over the original approach in which every ant build its own individual result set and later it is on built into a complete solution. However, in this approach the ants continuously modify a single result set instead of updating their own result set. Further, as it is cleared that a cloud is the collection of many nodes, which is able to support different types of application that is used by the clients on a basis of pay per use. Therefore, the system, which is incurring a cost for the user should function smoothly and also should have algorithms that are able to continue the proper system functioning at the peak usage hours.

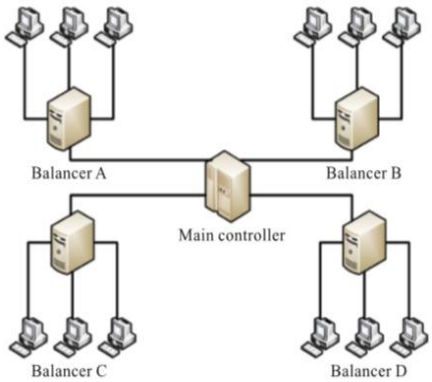

Fig. 1: Relationships between the main controllers, the balancers, and the nodes.

But in the cloud computing the data is stored as distributed manner. It is saved on the remote location or virtual locations randomly. If it is going to upload the data randomly on cloud it leads to an imbalance in cloud server storage. For example, some of the nodes are loaded with heavy functionality and other nodes are idle or doing very little work i.e. one server has been loaded by 10 GB data and second server has 0 or nigligible data uploaded on it.

3. Proposed Work

A Load Balancing Model for the Public Cloud which is based on cloud partitioning proposed efficient technique for balancing load in cloud. There is public cloud that has various nodes with distributed computing resources in number of different geographic locations. Thus, public cloud can be divided into several cloud partitions.

Whenever the environment is complex and huge, the load balancing can be simplified by these divisions. Then the he suitable partitions can be chosen by a main controller for arriving jobs. However, the balancer of each cloud partition chooses the best suitable load balancing strategy.

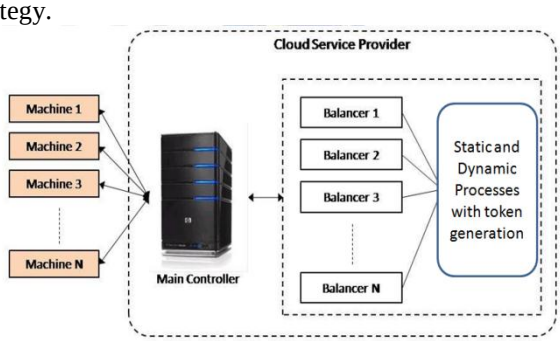

Fig. 2: System Architecture

A large public cloud has many nodes and those nodes can also be in different geographical locations. Cloud partitioning is one of the solutions to manage this large cloud. The public cloud has a subarea as a cloud partition with divisions based on the different geographic locations. When the creation the cloud partitions is carried out, a load balancing starts at the job arrival, with the main controller deciding which cloud partition should receive the individual job. The partition of load balancer decides how the jobs should be assigned to the nodes. The partitioning will be able to accomplish locally, when the load status of a cloud partition is normal. The job should be transferred to another different partition, if the cloud partition load status is not normal.

Application Servers:

- Cloud architecture may contain one to many number of application servers according to its scope and utility.
- Each application server has its number of dedicated resources

Master Servers:

- Master server is the first component which interacts with client and accept its request
- It divide task into number of activities

A Load Balancing Model Based on Cloud Partitioning for the Public Cloud proposed efficient technique for balancing load in cloud.

Define a load parameter set: F = {F1; F2;…; Fm} with each Fi($1 \leq i \leq m$; Fi $\in$ [0, 1]) parameter being either static or dynamic. m represents the total number of the parameters.

Then Compute the load degree as:

Load_degree(N) = $\sum_{i=1}^{m}$ aiFi

Load degree$_{avg}$ = $\frac{\sum_{i=1}^{m} Load\_Degree(Ni)}{n}$

Step 1 :

Get size of all Servers S1…..Sn

$S = \sum_{i=1}^{s} aiSi$

Step 2:

When, if S ∈ Server load(sL) is overflowed (limit exceeds) then

File fu uploads on S ∈ server next to sL.

Where

1) Load is Idle When

Load degree.(N) = 0;

2) Load is Normal when

0 < Load degree (N) ≤ Load degree$_{high}$

3) Load is Overload when

Load_degree high ≤ Load_degree(N)

In the currently proposed system the size of the server has not been set and also it is not taken as a parameter while uploading the load to the server so there might be a risk of overflowing the server. To avoid this, following contribution we will work which will help in building efficient system.

It may possible that one of the servers exceeds its size beyond the limit and also runs out of available bandwidth. The application will check the available space and actual size of the server. It may vary from each other and also the load of the server it contains so. After uploading the file it will check the size of file and the file will be uploaded to the other server if its size is greater than the available size. In this way, one can prevent Overflow of the server.

Token Generation

1. User U1 sends file to server for file uploading
2. Check balancers load b1 and b2

While(DataReader.read())
If (B1= DataReader[S])
Begin
Get load From server S[]={s1,s2,s3,…,sn}
End
Else
While(DataReader.read())
If(B2= DataReader[S])
Begin
Get load From server S[]={s1,s2,s3,…,sn}
End
Else

3. Generation of token.
4. After generating token we will implement token generation algorithm.
5. And using this algorithm we will upload the file and balance the server.
6. If File Size Data **FS** of user **U1** > available space of Server

Steps given below from a to c.

Begin

a) Prevent overflow of server S1….Sn that exceed a limit of size.
b) Perform load evaluation technique.
c) Upload file to the next server.

End

7. If File Size Data **FS** of user **U1** < available space of Server

Steps given below from a to b.

8.

Begin

a) Perform load evaluation technique.
b) Upload file to this server.

End

Conclusions and Future Scope

The system will get the load of all cloud systems to perform following functions as the client while Controller will get the load of all balancers and send it to the user by establishing connection to balancers. After this, by making connection to servers, balancer gets the load of all servers and sends it to the Controller.

Client will upload the files on the cloud by using load balancing algorithm to controller who uploads the file to the balancer who has a minimum load. Next, balancer uploads the file to the server that has a minimum load. Client can Search and Download the file and Deployments from the controller who will search and download the file from balancers. Next, cloud will Search and Download the particular file from sever.

Consideration of Static and dynamic parameters can support the load balancing based on cloud partitioning to improve the efficiency. But the consideration of dynamic processes can provide advanced load balancing over a cloud.

Some applications with the objectives of load balancing or reducing the operator's cost of service delivery are also feasible, but require further investigations to ensure efficiency, operational stability and overhead control. In light of the emerging efforts towards providing more security to the system, the latter application can become more pressing.

Acknowledgements

The authors would like to be thankful towards the anonymous reviewers for their valuable comments and suggestions.

References

Gaochao Xu, Junjie Pang, and Xiaodong Fu (2013), A Load Balancing Model Based on Cloud Partitioning for the Public Cloud Proc. 14th European Conf. Research in Computer Security (ESORICS '09 IEEE transactions on cloud computing year

K. Nishant, P. Sharma, V. Krishna, C. Gupta, K. P. Singh, N. Nitin, and R. Rastogi (Mar. 2012), Load balancing of nodes in cloud using ant colony optimization, in Proc. 14th International Conference on Computer Modelling and

Simulation (UKSim), Cambridgeshire, United Kingdom,, pp. 28-30.

Z. Chaczko, V. Mahadevan, S. Aslanzadeh, and C. Mcdermid (2011), Availability and load balancing in cloud computing,; presented at the 2011 International Conference on Computer and Software Modeling, Singapore.

Stavros Papadopoulos, Spiridon Bakiras, and Dimitris Papadias, pCloud: A Distributed System for Practical PIR , supported by grant HKUST 618108 from Hong Kong RGC, and by the NSF Career Award IIS- 0845262.

B. Adler, Load balancing in the cloud: Tools, tips and techniques, http://www.rightscale. com/info center/white-papers/Load-Balancing-in-the-Cloud.pdf, 2012

K. Ren, C. Wang, and Q. Wang (2012), Security Challenges for the Public Cloud, IEEE Internet Computing, vol. 16, no. 1, pp. 69-73

M. D. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis, and A. Vakali (Sept.-Oct. 2009), Cloud computing: Distributed internet computing for IT and scientific research, Internet Computing, vol.13, no.5, pp.10-13.

Grosu, A. T. Chronopoulos, and M. Y. Leung (Apr. 2002), Load balancing in distributed systems: An approach using cooperative games, in Proc. 16th IEEE Intl. Parallel and Distributed Processing Symp., Florida, USA, pp. 52-61.