

# Cloud computing load balancing model with Heterogeneous partition for public cloud

**Prathi Mani**

IV B.Tech (CSE)

*Department of Computer Science & Engineering  
CMR Technical Campus, India*

**Chette Kavitha**

IV B.Tech (CSE)

*Department of Computer Science & Engineering  
CMR Technical Campus, India*

**Sanganamoni Pallavi**

IV B.Tech (CSE)

*Department of Computer Science & Engineering  
CMR Technical Campus, India*

**MD. Rafeeq**

*Supervisor: Associate Professor*

*Department of Computer Science & Engineering  
CMR Technical Campus, India*

## Abstract

Load balancing in the cloud computing environment has an important impact on the performance. 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 for different situations. The algorithm applies the game theory to the load balancing strategy to improve the efficiency in the public cloud environment.

**Key words: load balancing model; public cloud; cloud partition; game theory**

## I. INTRODUCTION

Cloud computing is an attracting technology in the field of computer science. In Gartner's report, it says that the cloud will bring changes to the IT industry. The cloud is changing our life by providing users with new types of services. Users get service from a cloud without paying attention to the details. NIST gave a definition of cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. More and more people pay attention to cloud computing. Cloud computing is efficient and scalable but maintaining the stability of processing so many jobs in the cloud computing environment is a very complex problem with load balancing receiving much attention for researchers. Since the job arrival pattern is not predictable and the capacities of each node in the cloud differ, for load balancing problem, workload control is crucial to improve system performance and maintain stability. Load balancing schemes depending on whether the system dynamics are important can be either static and dynamic[6]. Static schemes do not use the system information and are less complex while dynamic schemes will bring additional costs for the system but can change as the system status changes..

## II. LITERATURE SURVEY

There have been many studies of load balancing for the cloud environment. Load balancing in cloud computing was described in a white paper written by Adler who introduced the tools and techniques commonly used for load balancing in the cloud. However, load balancing in the cloud is still a new problem that needs new architectures to adapt to many changes. Chaczkto et al described the role that load balancing plays in improving the performance and maintaining stability. There are many load balancing

algorithms, such as Round Robin, Equally Spread Current Execution Algorithm, and Ant Colony algorithm. Nishant et al. used the ant colony optimization method in nodes load balancing. Randles et al. gave a compared analysis of some algorithms in cloud computing by checking the performance time and cost. They concluded that the ESCE algorithm and throttled algorithm are better than the Round Robin algorithm. Some of the classical load balancing methods are similar to the allocation method in the operating system, for example, the Round Robin algorithm and the First Come First Served (FCFS) rules. The Round Robin algorithm is used here because it is fairly simple.

There are several cloud computing categories with this work focused on a public cloud. A public cloud is based on the standard cloud computing model, with service provided by a service provider. A large public cloud will include many nodes and the nodes in different geographical locations. Cloud partitioning is used to manage this large cloud. A cloud partition is a subarea of the public cloud with divisions based on the geographic locations.

The load balancing strategy is based on the cloud partitioning concept.

The partition load balancer then decides how to assign the jobs to the nodes. When the load status of a cloud partition is normal, this partitioning can be accomplished locally. If the cloud partition load status is not normal, this job should be transferred to another partition.

### III. SYSTEM ANALYSIS

#### A. Existing System

Cloud computing is efficient and scalable but maintaining the stability of processing so many jobs in the cloud computing environment is a very complex problem with load balancing receiving much attention for researchers. Since the job arrival pattern is not predictable and the capacities of each node in the cloud differ, for load balancing problem, workload control is crucial to improve system performance and maintain stability. Load balancing schemes depending on whether the system dynamics are important can be either static and dynamic. Static schemes do not use the system information and are less complex while dynamic schemes will bring additional costs for the system but can change as the system status changes. A dynamic scheme is used here for its flexibility.

#### B. Limitations

Cloud Computing environment is a very complex problem. The job arrival pattern is not predictable and the capacities of each node in the cloud differ. Workload control is crucial to improve system performance and maintain stability.

#### C. Proposed System

Load balancing schemes depending on whether the system dynamics are important can be either static and dynamic. Static schemes do not use the system information and are less complex while dynamic schemes will bring additional costs for the system but can change as the system status changes. A dynamic scheme is used here for its flexibility. The model has a main controller and balancers to gather and analyze the information. Thus, the dynamic control has little influence on the other working nodes. The system status then provides a basis for choosing the right load balancing strategy.

The load balancing model given in this article is aimed at the public cloud which has numerous nodes with distributed computing

resources in many different geographic locations. Thus, this model divides the public cloud into several cloud partitions. When the environment is very large and complex, these divisions simplify the load balancing. The cloud has a main controller that chooses the suitable partitions for arriving jobs while the balancer for each cloud partition chooses the best load balancing

#### D. Advantages

Mainly it divides the public cloud into several cloud partitions. When the environment is very large and complex these divisions simplify the load balancing. Load balancing improves the performance and maintain stability.

#### E. Software Requirements

- Operating system : Windows XP/7
- Coding Language : JAVA/J2EE
- Data Base : MYSQL

#### F. Hardware Requirements

- System : Pentium IV 2.4 GHz
- Hard Disk : 40 GB
- Floppy Drive : 1.44 Mb
- Monitor : 15 VGA Colour
- Mouse : Logitech
- Ram : 512 Mb

#### G. System Architecture

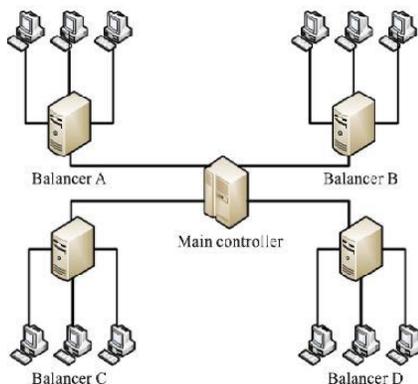


Fig 1: System Architecture

### IV. ODULES & DESCRIPTION

#### A. Modules

- User Module
- System Model
- Main Controller and Balancers
- Cloud Partition Load Balancing Strategy

#### B. Modules Description

### 1) *User Module*

In this module, Users are having authentication and security to access the detail which is presented in the ontology system.

Before accessing or searching the details user should have the account in that otherwise they should register first.

### 2) *System Module*

There are several cloud computing categories with this work focused on a public cloud. A public cloud is based on the standard cloud computing model, with service provided by a service provider . A large public cloud will include many nodes and the nodes in different geographical locations. Cloud partitioning is used to manage this large cloud. A cloud partition is a subarea of the public cloud with divisions based on the geographic locations. with the main controller deciding which cloud partition should receive the job. The partition load balancer then decides how to assign the jobs to the nodes. When the load status of a cloud partition is normal, this partitioning can be accomplished locally. If the cloud partition load status is not normal, this job should be transferred to another partition.

### 3) *Main Controller and Balancers*

The load balance solution is done by the main controller and the balancers. The main controller first assigns jobs to the suitable cloud partition and then communicates with the balancers in each partition to refresh this status information. Since the main controller deals with information for each partition, smaller data sets will lead to the higher processing rates. The balancers in each partition gather the status information from every node and then choose the right strategy to distribute the jobs.

### 4) *Cloud Partition Load Balancing Strategy*

When the cloud partition is idle, many computing resources are available and relatively few jobs are arriving. In this situation, this cloud partition has the ability to process jobs as quickly as possible so a simple load balancing method can be used. There are many simple load balance algorithm methods such as the Random algorithm, the Weight Round Robin, and the Dynamic Round Robin.

## V. IMPLEMENTATION

### A. *Cloud Computing Technology*

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers.

### B. *How Cloud Computing Works?*

The goal of cloud computing is to apply traditional supercomputing, or high-performance computing power, normally used by military and research facilities, to perform tens of trillions of computations per second, in consumer-oriented applications such as financial portfolios, to deliver personalized information, to provide data storage or to power large, immersive computer games.

### C. *Cloud Architecture*

The following diagram shows the major components of the Cloud operating system. Each section is described in more detail.

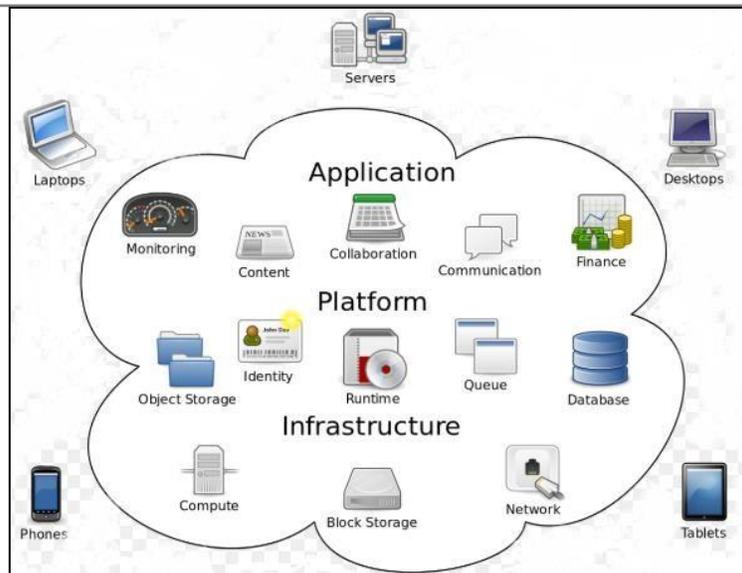


Fig. 2: Cloud Computing

## VI. SYSTEM TESTING

### A. Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

### B. Integration Testing

Integration test are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems.

### C. Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

- Valid Input : identified classes of valid input must be accepted.
- Invalid Input: identified classes of invalid input must be rejected.
- Functions : identified functions must be exercised.
- Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business

process flows; data fields, predefined.

#### ***D. System Testing***

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

#### ***E. White Box Testing***

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

#### ***F. Black Box Testing***

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot —see|| into it. The test provides inputs and responds to outputs without considering how the software works.

### **VII. FUTURE ENHANCEMENT**

Since this work is just a conceptual framework, more work is needed to implement the framework and resolve new problems. Some important points are:

- (1) Cloud division rules: Cloud division is not a simple problem. Thus, the framework will need a detailed cloud division methodology. For example, nodes in a cluster may be far from other nodes or there will be some clusters in the same geographic area that are still far apart. The division rule should simply be based on the geographic location (province or state).
- (2) How to set the refresh period: In the data statistics analysis, the main controller and the cloud partition balancers need to refresh the information at a fixed period. If the period is too short, the high frequency will influence the system performance. If the period is too long, the information will be too old to make good decision. Thus, tests and statistical tools are needed to set a reasonable refresh periods

### **VIII. CONCLUSION**

In the data statistics analysis the main controller and the cloud partition balancers need to revive the information at a fixed period. If the period is too short the high occurrence will pressure the system performance. If the period is too long the information will be too old to make good decision. Thus tests and statistical tools are needed to set sensible refresh periods. Other load balance strategies may provide better results so tests are essential to evaluate different strategies. Many tests are needed to assurance system accessibility and competence.

### **ACKNOWLEDGMENT**

We take this opportunity to express my profound gratitude and deep regard to our guide Mr. Md. Rafeeq Assoc. Professor for his

exemplary guidance, monitoring and constant encouragement throughout the project and paper work. We are also thankful to the Head of the Department Dr. K. Srujan Raju for providing excellent infrastructure and a nice atmosphere for completing this project successfully. We are obliged to our Director Dr. A. Raji Reddy for being cooperative throughout the course of this project. The guidance and support received from all the members of CMR Technical Campus who contributed and who are contributing to this project, was vital for the success of the project. We are grateful for their constant support and help.

#### REFERENCES

- [1] P. Mell and T. Grance, -The nist definition of cloud computing, <http://dx.doi.org/10.602/NIST.SP.800-145>.
- [2] 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.
- [3] [1] R. Hunter, The why of cloud, [http://www.gartner.com/DisplayDocument?doc\\_cd=226469&ref=gnoreg](http://www.gartner.com/DisplayDocument?doc_cd=226469&ref=gnoreg), 2012.
- [4] [2] M. D. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis, and A. Vakali, Cloud computing: Distributed internet computing for IT and scientific research, *Internet Computing*, vol.13, no.5, pp.10-13, Sept.-Oct. 2009.
- [6] [3] P. Mell and T. Grance, The NIST definition of cloud computing, <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>, 2012.
- [7] [4] Microsoft Academic Research, Cloud computing, <http://libra.msra.cn/Keyword/6051/cloud-computing?query=cloud%20computing>, 2012.
- [8] [5] Google Trends, Cloud computing, <http://www.google.com/trends/explore#q=cloud%20computing>, 2012.
- [9] [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.
- [10] [7] 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
- [11] [8] Z. Chaczko, V. Mahadevan, S. Aslanzadeh, and C. Mcdermid, Availability and load balancing in cloud computing, presented at the 2011 International Conference on Computer and Software Modeling, Singapore, 2011.
- [12] [9] 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

---

Modelling and Simulation (UKSim), Cambridgeshire, United Kingdom, Mar. 2012, pp. 28-30.

- [13] [10] 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, pp. 551-556.
- [14] [11] A. Rouse, Public cloud, <http://searchcloudcomputing.techtarget.com/definition/public-cloud>, 2012.
- [15] [12] D. MacVittie, Intro to load balancing for developers — The algorithms, <https://devcentral.f5.com/blogs/us/introto-load-balancing-for-developers-ndash-the-algorithms>, 2012.
- [16] [13] 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.
- [17] [14] D. Grosu, A. T. Chronopoulos, and M. Y. Leung, Load balancing in distributed systems: An approach using cooperative games, in Proc. 16th IEEE Intl. Parallel and Distributed Processing Symp., Florida, USA, Apr. 2002, pp. 52-61.
- [18] [15] S. Aote and M. U. Kharat, A game-theoretic model for dynamic load balancing in distributed systems, in Proc. the International Conference on

