

# An Overview on Performance Issues in Cloud Computing

Aswathi Vandana P.  
PG Scholar  
Sri Ramakrishna Engineering College,  
Tamil Nadu, India

Nandhini A.  
PG Scholar  
Sri Ramakrishna Engineering College,  
Tamil Nadu, India

Saravana Balaji B.  
Assistant Professor  
Sri Ramakrishna Engineering College,  
Tamil Nadu, India

Dr.N.K.Karthikeyan  
Prof. & Head-IT  
Srikrishna College of Engg. & Technology,  
Tamil Nadu, India

## Abstract

Cloud computing provides huge computing services to the business for improving the organizational growth. It adopts the concept of virtualization, service oriented architecture, autonomic, and utility computing. The cloud has many advantages and it is easy to implement with any business logics. Cloud delivers services from different data sources and servers located on different geographical location but the user gets single point of view from the cloud service. As advancements of various areas of technology increases, different types of issues have been introduced in cloud. In this paper, we survey about the various types of issues and challenges associated with cloud computing especially performance issues and cloud storage security issues. Cloud computing saves time, money and effort. Finally, the paper also presents a brief discussion about various strategies of improvisation of performance in cloud..

**Keyword:** Cloud Computing, Reliability and Fault tolerance, Load balancing.

## 1. Introduction

Cloud is an emerging technology where the providers are provide various services to mainly IT sectors. Cloud computing is a model for enabling a convenient, on demand network access to a shared pool of configurable computing resources (e.g., network, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Cloud computing provides an online storage, that is used to store large amount of data and we can access data where ever we are. No need to carry any physical device with us is the main advantage of Cloud computing. Due to multi-tenancy, there are many risks for cloud storage such as confidential data, integrity of data and modification of data.

## 1.1 Service Architecture of Cloud Computing

NIST[1] defines three main service models for cloud computing:

- **Software as a Service (SaaS)** – The cloud provider provides the cloud consumer with the capability to deploy an application on a cloud infrastructure
- **Platform as a Service (PaaS)**– The cloud provider provides the cloud consumer with the capability to develop and deploy applications on a cloud infrastructure using tools, runtimes, and services supported by the CSP.
- **Infrastructure as a Service (IaaS)** – The cloud provider provides the cloud consumer with essentially a virtual machine. The cloud consumer has the ability to provision processing, storage, networks, etc., and to deploy and run arbitrary software supported by the operating system run by the virtual machine.

## 1.2 Deployment models of Cloud Computing

Cloud Computing has four main deployment models from the architecture, each with specific characteristics that supports the needs of the services and users of the clouds in particular ways [2].

### 1. Private Cloud:

Private clouds are owned and operated by a user or a cloud computing provider, this type of cloud is built for the sole use of a single user. Private clouds utilize the same technology as public clouds and it's mainly built to enable an individual company to maximize the use of its computing resources and be more responsive to company needs.

### 2. Public Cloud:

Public clouds are owned and operated by third parties and located in data centers that operate outside of the user location. Multiple companies share these resources; each cloud user is assigned own virtual computing capabilities based on a common set of physical resources. Public clouds are provided by

companies like Amazon, Hewlett-Packard, IBM, Google, Microsoft, Rackspace, Salesforce.com.

### 3. Hybrid Cloud:

Hybrid clouds are combinations of multiple clouds that are both public and private. These clouds are created by individual customers to meet their precise needs. For example, a company may decide to create a hybrid cloud to combine a CRM system provided on a public cloud operated by Salesforce.com with an ERP system running on their private cloud.

### 4. Community Cloud:

Community cloud infrastructure is shared by some organizations and supports a specific community that shares concerns (e.g., mission, security requirements, and govt. education and compliance considerations). It may either be managed by the organizations or a third party and may exist on premise or off premise.

## 2. Issues in Cloud Computing

There are various types of issues in Cloud Computing as the technology proceeds. The following are the issues and the solutions that had been proposed concerned with the issues.

### 2.1 Issues in Performance

Performance issue deals with the various challenges in acquiring reliable, fault tolerant quality services. It also handles the problems arising due to inefficient load balancing.

#### 2.1.1 Challenge in Reliability and Fault Tolerance

There are many challenges in cloud computing that deal with the building of highly reliable complex applications on distributed resources in large scale and since there are wide offerings, appropriate selection of cloud services as per the requirement, is becoming hard. In this survey paper, we describe two frameworks regarding reliability and fault tolerance.

##### 2.1.1a) BFTCloud:

BFTCloud is a Byzantine Fault Tolerance framework [3] for building robust systems in voluntary-resource cloud environments.

In general, the reliability of cloud applications is greatly influenced by the reliability of cloud modules. This paved the way to build high-reliable cloud applications. To build reliable cloud applications on the voluntary-resource cloud infrastructure, it is extremely critical to design a fault

tolerance mechanism for handling several faults that includes node faults like crashing, network faults like disconnection, Byzantine faults [4] like malicious behaviors, etc. To focus the critical challenge, we propose an innovative approach, called Byzantine Fault Tolerant Cloud (BFT Cloud), for tolerating different types of failures in voluntary resource clouds.

BFTCloud employs replication techniques for overwhelming failures. BFTCloud can also be integrated into cloud nodes as a middleware.

The following Figure 1 shows the system architecture of BFTCloud in voluntary-resource cloud environment.

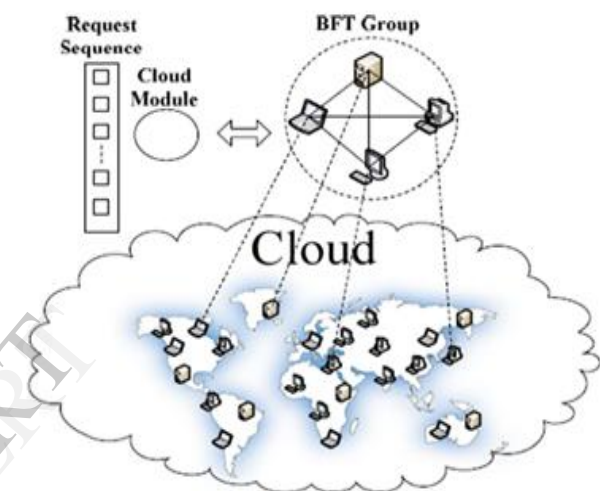


Figure 1: Architecture of BFTCloud in Voluntary-Resource Cloud

Cloud application consists of many cloud modules. In order to guarantee the robustness of the module, it must choose a BFT group from the pool of cloud nodes for request execution. In addition, a monitor is implemented on the cloud module side as a middleware for monitoring the QoS performance and failure probability of nodes.

The following Figure 2 shows the work procedures of BFTCloud. The input of BFTCloud is a sequence of requests with specified QoS requirements sent by the cloud module. The output of BFTCloud is a sequence of committed responses corresponding to the requests.

BFTCloud consists of five phases described as below:

#### 1. Primary Selection:

The primary is selected by applying the primary selection algorithm with respect to the QoS requirements of the request.

**2. Replica Selection:**

In this phase, a set of nodes are selected as replicas by applying a replica selection algorithm with respect to the QoS requirements of the request. The primary then forwards the request to all replicas for execution.

**3. Request Execution:**

In this phase, all members in the BFT group execute the request locally and send back their responses to the cloud module.

**4. Primary Updating:**

In this phase, faulty primary in the BFT group will be identified and replaced by a newly selected primary.

**5. Replica Updating:**

In this phase, the replica updating algorithm will be applied to replace the faulty replicas with other suitable nodes.

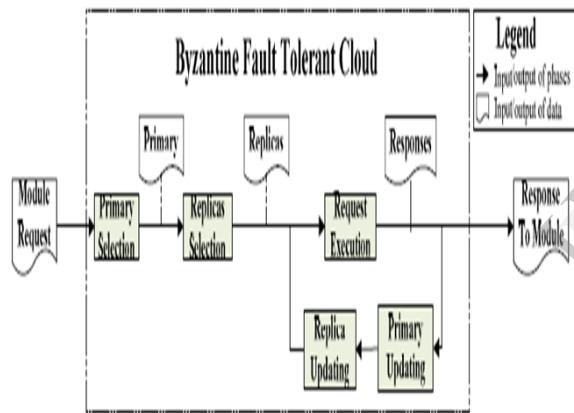


Figure 2: Work procedures of BFTCloud

The experimental results show that BFTCloud guarantees high reliability of systems built on the top of voluntary-resource cloud infrastructure and ensures good performance of these systems.

**2.1.1 b) SMICloud:**

At present, there is no software framework which can automatically index cloud providers based on the customer requirements, so this might lead to an expensive usage of resources. This paper deals with a framework and mechanisms, which measure the quality and prioritize cloud services. Hence customer can decide about the apt services that satisfy SLA and improved QoS based on their needs.

SMICloud [5] is based on Service Measurement Index (SMI) provided by Cloud Service Measurement Index Consortium (CSMIC) [6], since its considered to be vital for the evaluation of cloud services. There are several challenges faced in evaluating QoS and ranking Cloud providers [7]. The first is how to measure various SMI attributes, since many of these attributes vary over time. The second challenge is deciding which service matches the best based on SMI attribute

In the next section, an overview of SMI and its high level QoS Attributes has been presented. It's followed by SMICloud framework with its key components. And how metrics for various qualities attributes can be modeled. Later, the Cloud ranking mechanism which is explained by case study example. Finally it concludes with some future works.

SMI attributes are designed based on International Organization for Standardization (ISO) standards by the Consortium. SMI framework provides a clear view of QoS needed by the customers for selecting a Cloud service provider based on: Accountability, Agility, Assurance of Service, Cost, Performance, Security and Privacy, and Usability.

The Service Measurement Index Cloud framework- SMICloud has been proposed, to help Cloud customers to find the most suitable Cloud provider and therefore can initiate SLAs. In addition, SMICloud framework provides features such as service selection based on Quality of Service (QoS) requirements and ranking of services based on previous user experiences and performance of services.

The following figure, Figure 3 shows the key elements of the SMICloud framework:

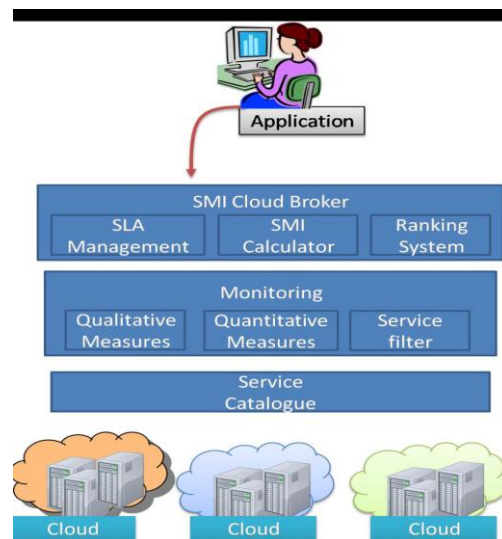


Figure 3 SMICloud Framework

### 1. SMICloud Broker

It receives the request for deployment of an application from the customer. And also collects all the requirements from the customer and performs the discovery and ranking of suitable services using other components.

### 2. Monitoring

Monitoring component first discovers the Cloud services which can satisfy user's essential Quality of Service requirements. Thus it monitors the performance of Cloud services.

### 3. Service Catalogue

It stores the services and their features advertised by various Cloud providers.

#### 2.1.2 Challenge In Load Balancing

Load balancing is dividing the amount of work that a computer has to do between two or more computers so that more work gets done in the same amount of time and, in general, all users get served faster as shown in Figure 4 . Load balancing can be implemented with hardware, software, or a combination of both. Load balancing optimizes resource use, maximizes throughput, minimizes response time, and avoids overload [8]. Using multiple components with load balancing, instead of a single component, may increase reliability through redundancy. When you apply load balancing during runtime, it is called dynamic load balancing — this can be realized both in a direct or iterative manner according to the execution node selection.

- In the iterative methods, the final destination node is determined through several iteration steps.
- In the direct methods, the final destination node is selected in one step. These approaches aim to enhance the overall performance of the Cloud and provide the user more satisfying and efficient services.

#### Goals of Load Balancing

Goals of load balancing are discussed by authors [9] include:

- Significant improvement in performance
- Maintenance of the system stability.
- Increase flexibility of the system.
- Build a fault tolerant system by creating backups.

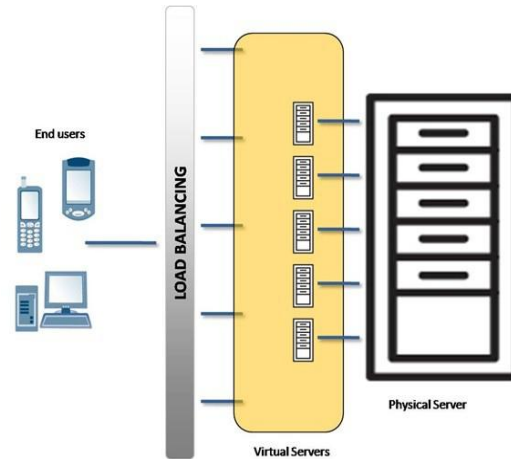


Figure 4: Load Balancing in Cloud

#### 2.1.2 a) Ant colony optimization (ACO):

A cloud is constituted by various nodes which perform computation according to the requests of the clients. As the requests of the clients can be random to the nodes they can vary in quantity and thus the load on each node can also vary. Therefore, every node in a cloud can be unevenly loaded of tasks according to the amount of work requested by the clients. This phenomenon can drastically reduce the working efficiency of the cloud as some nodes which are overloaded will have a higher task completion time compared to the corresponding time taken on an under loaded node in the same cloud. This problem is not only confined only to cloud but is related with every large network like a grid, etc. we propose an efficient algorithm, based on ACO for better distribution of workload among the nodes of a cloud.

The ant uses the basic pheromone updating formula and node selection formula of the ACO to distribute evenly the work loads of nodes in a cloud. For efficient load balancing of work in cloud, tier-wise distribution of nodes is also suggested [10], in this the nodes are distributed in three tier structure such that the work is properly distributed among the nodes. In this hierarchy the 1st level (Top-level) nodes are used for the proper distribution of work among the nodes of 2nd level. Simultaneously the 2nd level distributes the work logically among the 3<sup>rd</sup> level nodes, which in turn-process their part of work. Thus, this system ensures the proper distribution of load among all levels.

For building an optimum solution set. We first select a Regional load balancing node (RLBN) is chosen in a CCSP, which will act as a head node. We would be referring to the RLBN as head node in the rest. The selection of head node is not a permanent



thing but a new head node can be elected if the previous node stops functioning properly due to some inevitable circumstances. The head node is chosen in such way that it has the most number of neighboring nodes, as this can help our ants to traverse in most possible directions of the network of CCSP [11].

These ants traverse the width and length of the network in such a way that they know about the location of under loaded or overloaded nodes in the network. These Ants along with their traversal will be updating a pheromone table, which will keep a tab on the resources utilization by each node. We also proposed the movement of ants in two ways similar to the classical ACO, which are as follows:

- Forward movement-The ants continuously move in the forward direction in the cloud encountering overloaded node or under loaded node.
- Backward movement-If an ant encounters an overloaded node in its movement when it has previously encountered an under loaded node then it will go backward to the under loaded node to check if the node is still under loaded or not and if it finds it still under loaded then it will redistribute the work to the under loaded node.

The main benefit of this approach lies in its detections of overloaded and under loaded nodes and thereby performing operations based on the identified nodes. This simplistic approach elegantly performs our task of identification of nodes by the ants and tracing its path consequently in search of different types of nodes. The ants continuously update a single result set rather than updating their own result set. In this way, the solution set is gradually built on and continuously improved upon rather than being compiled only once in a while.

### 2.1.2 b) Load balancing algorithm in VMcloud:

Load balancing is one of prerequisites to utilize the full resources of parallel and distributed systems. Load balancing mechanisms can be broadly categorized as centralized or decentralized, dynamic or static, and periodic or non-periodic. Physical resources can be split into a number of logical slices called Virtual Machines (VMs). All VM load balancing methods are designed to determine which Virtual Machine assigned to the next cloudlet [8].

DataCenter object manages the data center management activities such as VM creation and destruction and does the routing of user requests received from User Bases via the Internet to the VMs. The Data Center Controller [12], uses a VmLoadBalancer to determine which VM should be assigned the next request for processing. Most common Vmloadbalancer are throttled and active monitoring load balancing algorithms.

### 1. Throttled Load Balancer:

It maintain a record of the state of each virtual machine (busy/ ideal), if a request arrive throttled load balancer send the ID of ideal virtual machine to the data center controller and it allocates the ideal virtual machine.

### 2. Active Monitoring Load Balancer:

Active VM Load Balancer maintains information about each VMs and the number of requests currently allocated to which VM. When a request arrives, it identifies the least loaded VM. If there are more than one, the first identified is selected. Data Center Controller notifies the Active Vm Load Balancer of the new allocation.

The Proposed Load balancing algorithm is divided into three parts. The first phase is the initialization phase. In the first phase, the expected response time of each VM is to be found. In second Phase find the efficient VM, in Last Phase return the ID of efficient VM.

- Efficient algorithms find expected response time of each Virtual machine.
- When a request to allocate a new VM from the Data Center Controller arrives, Algorithms find the most efficient VM for allocation.
- Efficient algorithms return the id of the efficient VM to the Datacenter Controller.
- Datacenter Controller notifies the new allocation
- Updates the allocation table increasing the allocations count for that VM.
- When the VM finishes processing the request and the Data Center Controller receives the Response. Data center controller notifies the efficient algorithm for the VM de-allocation.

We conclude that if we select an efficient virtual machine then it affect the overall performance of the cloud Environment and also decrease the average response time is decrease.

### 2.1.2 c) Dynamic load balancing:

Dynamic load balancing algorithms, the current state of the system is used to make any decision for load balancing. It allows

For processes to move from an over utilized machine to an under-utilized machine dynamically for faster execution. This means that it allows for process preemption which is not supported in Static load balancing approach. An important advantage of this approach is that its decision for balancing the load is based on the current state of the system which helps in

improving the overall performance of the system by migrating the load dynamically [13].

• **Dynamic Load Balancing Policies or Strategies:**

The different policies as described in [14] are as follows:

**1. Location Policy:**

The policy used by a processor or machine for sharing the task transferred by an over loaded machine is termed as Location policy.

**2. Transfer Policy:**

The policy used for selecting a task or process from a local machine for transfer to a remote machine is termed as Transfer policy.

**3. Selection Policy:**

The policy used for identifying the processors or machines that take part in load balancing is termed as Selection Policy.

**4. Information Policy:**

The policy that is accountable for gathering all the information on which the decision of load balancing is based is referred as Information policy.

**5. Load estimation Policy:**

The policy which is used for deciding the method for approximating the total work load of a processor or machine is termed as Load estimation policy.

**6. Process Transfer Policy:**

The policy which is used for deciding the execution of a task that is to be done locally or remotely is termed as Process Transfer policy.

**7. Priority Assignment Policy:**

The policy that is used to assign priority for execution of both local and remote processes and tasks is termed as Priority Assignment Policy.

**8. Migration Limiting Policy:**

The policy that is used to set a limit on the maximum number of times a task can migrate from one machine to another machine.

**3. Conclusion:**

Concluding, Cloud computing is an emerging technology supports business and satisfies customer needs by providing on-demand services in a shared environment. Cloud computing is becoming a popular and important solution for building highly reliable applications on distributed resources. This paper mainly focuses on an overview of Cloud Computing along with the performance issues. Also we had presented a precise idea on reliability and fault tolerance by means of BFTCloud and SMICloud. It is followed by the various issues associated with load balancing along with the solutions.

**References:**

[1] Bhaskar Prasad Rimal, Enumi Choi, Ian Lumb, "A taxonomy and survey of cloud computing systems", 5th

International Joint Conference in INC, IMS and IDC, 978-0-7695-3769-6/09, 2009, pp 44-51.

[2] Qi Zhang · Lu Cheng · Raouf Boutaba, "Cloud computing: state-of-the-art and research challenges", J Internet Serv Appl, Springer, DOI 10.1007/s13174-010-0007-6, 2010, pp 7-18.

[3] Yilei Zhang, Zibin Zheng and Michael R. Lyu, "BFTCloud: A Byzantine Fault Tolerance Framework for Voluntary-Resource Cloud Computing", 2011 IEEE 4th International Conference on Cloud Computing.

[4] L. Lamport, R. Shostak, and M. Pease, "The Byzantine generals problem," *ACM Transactions on Programming Languages and Systems (TOPLAS)*, vol. 4, no. 3, pp. 382–401, 1982.

[5] Saurabh Kumar Garg, Steve Versteeg and Rajkumar Buyya, "SMICloud: A Framework for Comparing and Ranking Cloud Services", 2011 Fourth IEEE International Conference on Utility and Cloud Computing.

[6] C.S.M.I.C. (CSMIC), "SMI Framework," URL:<http://betawww.cloudcommons.com/servicemeasureme ntindex>.

[7] J. Varia, *Cloud Computing: Principles and Paradigms*. Wiley Press, 2011, ch. 18: Best Practices in Architecting Cloud Applications in the AWS Cloud, pp. 459–490.

[8] R. Shinmonski. *Windows 2000 & Windows Server 2003." Clustering and Load balancing"*. Emeryville. McGraw-Hill professional publishing, CA, USA (2003), p2, 2003.

[9] David Escalante and Andrew J. Korty, "Cloud Services: Policy and Assessment", *EDUCAUSE Review*, Vol. 46, July/August 2011.

[10] S.C. Wang, K.Q. Yan, W.P. Liao and S.S. Wang, "Towards a Load Balancing in a Three-level Cloud Computing Network", *Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology*, pp. 108-113, 2010.

[11] Kumar Nishant, Pratik Sharma, Vishal Krishna, Chhavi Gupta and Kunwar Pratap Singh "Load Balancing of Nodes in Cloud Using Ant Colony Optimization", *proceedings of 14th International Conference on Modelling and Simulation*.

[12] Bhatiya Wickremasinghe, Rodrigo N. Calheiros, Rajkumar Buyya, "CloudAnalyst: A CloudSim-based Visual Modeller for Analysing Cloud Computing Environments and Applications", 20-23, April 2010, pp. 446-452.

[13] Meenakshi Sharma, Pankaj Sharma, Dr. Sandeep Sharma, "Efficient Load Balancing Algorithm in VM Cloud Environment", *IJCST Vol. 3, Issue 1, Jan. - March 2012*.

[14] Abhijit A Rajguru, S.S. Apte, "A Comparative Performance Analysis of Load Balancing Algorithms In Distributed Systems Using Qualitative Parameters", *International Journal of Recent Technology and Engineering*, Vol. 1, Issue 3, August 2012.