LOAD BALANCING IN PUBLIC CLOUD BY DIVISION OF CLOUD BASED ON THE GEOGRAPHICAL LOCATION

Abhijeet G Purohit[1], Md. Abdul Waheed[2], Asma Parveen[3]

[1]MTech (CSE) Student, Computer Science and Engg Dept., KBN College of Engg, Gulbarga, Karnataka, India
[2]Professor, Computer Science and Engg Dept, VTU Regional Office, Gulbarga, Karnataka, India
[3]HOD, Computer Science and Engg Dept., KBN College of Engg, Gulbarga, Karnataka, India

Abstract

Load balancing is a method of controlling the traffic in a cloud environment. Cloud applications look for resources for execution. The resources can be storage, processing, bandwidth, etc. Allocation of these resources efficiently to all the competing jobs is called as load balancing. In this paper, we describe load balancing in a public cloud by partitioning the cloud into several sub-clouds. This division of public cloud into several sub-clouds is done based on the geographical location. In this approach we use a central controlling system that monitors all the sub clouds. Here, every sub cloud has a balancer system which monitors the resources in its sub cloud and allocates the available resources to the competing jobs. These balancer systems also communicate with the central controlling system about the status of the respective sub cloud. Based on this information the central controlling system selects the optimal sub cloud.

Keywords: *load balancing, public cloud, jobs, overload, central controller system and balancers.*

----------***----------

1. INTRODUCTION

1.1 Definition

Cloud computing can be defined as a practice of using a network of remote servers hosted on the internet to store, manage, and process data , rather than a local server or a desktop. According to NIST, cloud computing is 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 [8].

1.2 Essential Characteristics of Cloud Computing

1. On-demand self-service: A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.
2. Broad network access: Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).
3. Resource pooling: The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand.
4. Rapid elasticity: Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.
5. Measured service: Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.

1.3 Service Models

Software as a Service (SaaS): The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

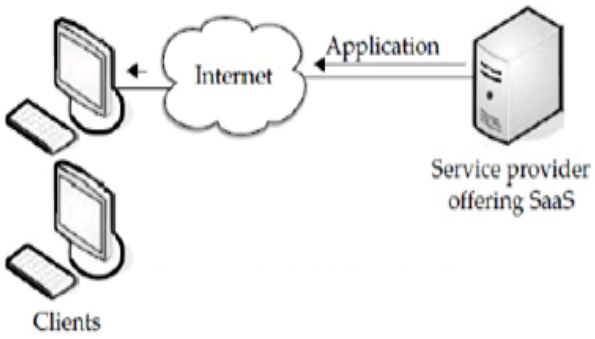

Fig -1: SaaS [21]

Platform as a Service (PaaS): The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

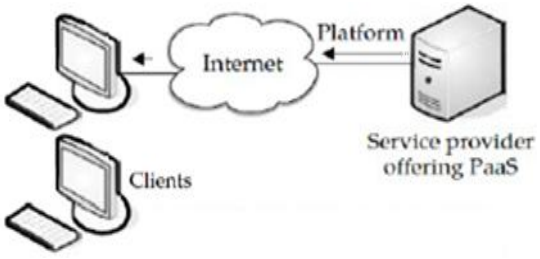

Fig -2: PaaS [21]

Infrastructure as a Service (IaaS): The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls).

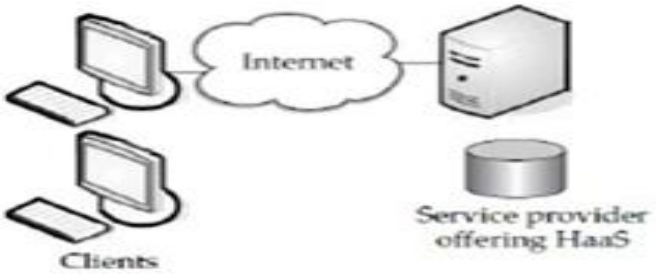

Fig -3: IaaS [21]

1.4 Deployment Models

Private cloud: The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.

Community cloud: The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.

Public cloud: The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider [13].

Hybrid cloud: The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds).

1.5 Virtualization

It is a very useful concept in the context of cloud systems. Virtualization means "something which isn't real", but gives all the facilities of a real system. It is the software implementation of a computer which will execute different programs like a real machine. Virtualization is related to cloud, because using virtualization an end user can use different services of a cloud. The remote datacenter will provide different services in full or partial virtualized manner. There are two types of virtualization found in cloud environment:

Full virtualization: In case of full virtualization a complete installation of one machine is done on the machine. It will result in a virtual machine which will have all the software that is present in the actual server.

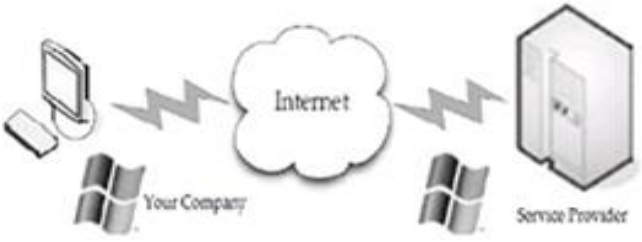

Fig -4: Full Virtualization [21]

Para virtualization: In Para-virtualization, the hardware allows multiple operating systems to run on single machine by efficient use of system resources such as memory, etc.

Fig -5: Para-Virtualization [21]

1.6 Load Balancing

Load Balancing is a method to distribute workload across one or more servers, network interfaces, hard drives, or other computing resources [1]. Typical datacenter implementations rely on large, powerful (and expensive) computing hardware and network infrastructure, which are subject to the usual risks associated with any physical device, including hardware failure, power and/or network interruptions, and resource limitations in times of high demand.

The load balancing service is usually provided by dedicated software or hardware, such as a multilayer switch or a Domain Name System server. Load balancing models and algorithms rely either on session-switching at the application layer, packet-switching at the network layer or processor load balancing mode [12]. Load balancing keep costs low, enterprise greener and puts less stress on resources thus making the resources last longer.

Depending on the particular application's architecture, technology stack, traffic patterns, and numerous other variables, there may be one or more viable schemes or solutions. They can be classified based on the system load or system topology [2]:

1. System Load: They are further classified as:

    a) Centralized Scheme: Here, a master node manages the entire system.
    b) Distributed Scheme: Here, all the nodes are independent of one another.

2. System Topology: They are further classified as:

    a) Static Scheme: These schemes do not use the system information when balancing the load.

Dynamic Scheme: Here, the current state of the node is crucial for balancing the load.

2. RELATED WORK

The article, "Virtual Infrastructure Management in Private and Hybrid Clouds," by Borja Sotomayor, Rubén S. Montero, Ignacio M. Llorente, and Ian Foster, presents two open source projects for private and hybrid clouds. OpenNebula is a virtual infrastructure manager that can be used to deploy virtualized services on both a local pool of resources and on external IaaS clouds. Haizea is a resource lease manager that can act as a scheduling back end for OpenNebula, providing advance reservations and resource preemption [5].

"Harnessing Cloud Technologies for a Virtualized Distributed Computing Infrastructure," by Alexandre di Costanzo, Marcos Dias de Assunção, and Rajkumar Buyya, presents the realization of a system termed the InterGrid for interconnecting distributed computing infrastructures by harnessing virtual machines. The article provides an abstract view of the proposed architecture and its implementation. Experiments show the scalability of an InterGrid-managed infrastructure and how the system can benefit from using cloud infrastructure [6].

In "Content-Centered Collaboration Spaces in the Cloud," John S. Erickson, Susan Spence, Michael Rhodes, David Banks, James Rutherford, Edwin Simpson, Guillaume Belrose, and Russell Perry envision a cloud-based platform that inverts the traditional application-content relationship by placing content rather than applications at the center, letting users rapidly build customized solutions around their content items. The authors review the dominant trends in computing that motivate the exploration of new approaches for content-centered collaboration and offer insights into how certain core problems for users and organizations are being addressed today [3].

In the article, "Sky Computing," by Katarzyna Keahey, Maurício Tsugawa, Andréa Matsunaga, and José A.B. Fortes, describes the creation of environments configured on resources provisioned across multiple distributed IaaS clouds. This technology is called sky computing. The authors provide a real-world example and illustrate its benefits with a deployment in three distinct clouds of a bioinformatics application [7].

3. SYSTEM MODEL

A public cloud is a set of computers and computer network resources based on the standard cloud computing model, in which a service provider makes resources, such as applications and storage, available over the Internet [13]. In this load balancing model the public cloud is partitioned based on their geographical locations. Figure 6 [20], below shows the schematic of a partitioned public cloud. 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 CCS that chooses

the suitable partitions for arriving jobs while the balancer for each cloud partition chooses the best load balancing strategy.

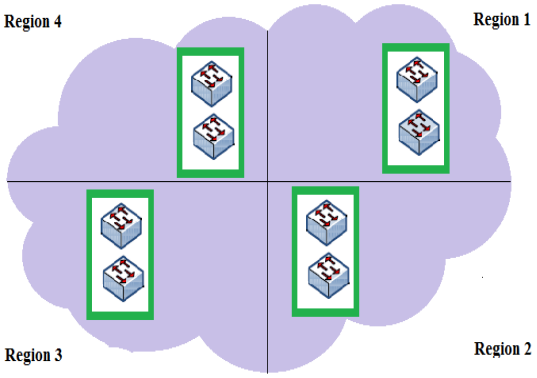

Fig -6: Schematic of Partitioned public cloud.

Load balancing is based on this partitioning of public cloud[4]. When the jobs arrive a best partition is selected and the jobs are processed by the servers present in that partition. This selection of best partition is done by a central module called as Central Controller System (CCS) and the distribution of jobs among the servers is done by the balancers present in every partition.

3.1 Central Controller System (CCS)

It is a kind of main controller which decides the sub partition to be selected in the public cloud. The goal of this module is to interact with the balancers and application servers (or nodes) and collect necessary status information of the system, this is shown in figure 7 [20]. The interaction between CCS and Balancer is a two-way interaction and that of between a CCS and an Apps server is a one-way interaction. Based on the collected information and necessary calculations made, a geographically nearest partition is selected. This selection of geographically nearest partition is to minimize or avoid the cost incurred on moving the jobs to a distant partition.

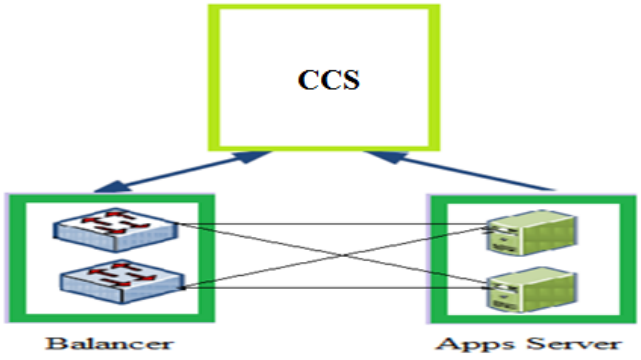

Fig -7: CCS collecting status info from balancer and apps server.

3.2 Balancer System (BS)

Balancers distribute the jobs to its application servers or nodes. Balancers also play a vital role in this model. Balancers can also identify the scope for parallel execution of jobs. However, there no single algorithm which can deal with different load balancing situation. Due to this different, algorithms can be used in different balancers for different situations and/or different workloads. Table below shows a list of existing algorithms which can be used in balancers.

Table -1: Comparison of algorithms [15], [18]

Metrics	Active clustering	OLB+ LBMM	Join Idle Queue	Min-min	Min-max
Throughput	No	No	No	Yes	Yes
Overhead	Yes	No	Yes	Yes	Yes
Fault tolerance	No	No	No	No	No
Migration Time	Yes	No	No	No	No
Response Time	No	No	Yes	Yes	Yes
Resource utilization	Yes	Yes	No	Yes	Yes
Scalability	No	No	No	No	No
Performance	No	Yes	Yes	Yes	Yes

Table -2: Comparison of algorithms [14], [16], [19]

Algorithm	Nature	Environment	Process Migration	Resource Utilization
Token Routing	Dynamic	Decentralized	Yes	More
Round Robin	Static	Decentralized	Yes	Less
Randomized	Static	Decentralized	No	Less
Central Queuing	Dynamic	Centralized	No	Less
Least Connection	Dynamic	Centralized	No	Less

4. CONCLUSIONS

Load balancing is a recurring problem. Due to this a dynamic solution is needed to balance the load. In this paper we have describe a framework which can accommodate multiple suitable scheduling algorithms based on the status of the balancer system and the workload. Table 1 and 2 gives a comparison between the algorithms which can best fit this frame work.

ACKNOWLEDGEMENTS

We sincerely thank all the staff of Computer Science and Engineering Department at KBN College of Engineering, Gulbarga, Karnataka.

REFERENCES

[1]. 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.
[2]. Amandeep Kaur Sidhu, Supriya Kinger, Analysis of Load Balancing Techniques in Cloud Computing presented at International Journal of Computers & Technology Volume 4 No. 2, March-April, 2013, ISSN 2277-3061.
[3]. John S. Erickson, Susan Spence, Michael Rhodes, David Banks, James Rutherford, Edwin Simpson, Guillaume Belrose, and Russell Perry, Content-Centered Collaboration Spaces in the Cloud presented at IEEE Internet Computing, volume 13 Issue 5, September 2009.
[4]. Gaochao Xu, Junjie Pang, and Xiaodong Fu, A Load Balancing Model Based on Cloud Partitioning for the Public Cloud presented at IEEE Transactions on Cloud Computing, 2013.
[5]. Borja Sotomayor, Rubén S. Montero, Ignacio M. Llorente, and Ian Foster, Virtual Infrastructure Management in Private and Hybrid Clouds, presented in IEEE Internet Computing Special Issue on Cloud Computing.
[6]. Alexandre di Costanzo, Marcos Dias de Assunção, and Rajkumar Buyya, Harnessing Cloud Technologies for a Virtu-alized Distributed Computing Infrastructure, presented in IEEE Computer Society during September/October 2009 (vol. 13 no. 5)
[7]. Katarzyna Keahey, Maurício Tsugawa, Andréa Matsunaga, and José A.B. Fortes, Sky Computing.
[8]. P. Mell and T. Grance, The NIST definition of cloud computing, http://csrc.nist.gov/publications/ nistpubs/800-145/SP800-145.pdf, 2012.
[12]. Branko Radojević, Mario Žagar, Analysis of Issues with Load Balancing Algorithms in Hosted (Cloud) Environments.
[13]. http://en.wikipedia.org/wiki/Public_cloud
[14]. Soumya Ray and Ajanta De Sarkar , Execution Analysis of Load Balancing Algorithms in Cloud Computing Environment in International Journal on Cloud Computing: Services and Architecture (IJCCSA),Vol.2, No.5, October 2012.
[15]. Nayandeep Sran and Navdeep Kaur, Comparative Analysis of Existing Load Balancing Techniques in Cloud Computing in International Journal of Engineering Science Invention Volume 2 Issue 1 January 2013.
[16]. Zhong Xu, Rong Huang,(2009)"Performance Study of Load Balanacing Algorithms in Distributed Web Server Systems", CS213 Parallel and Distributed Processing Project Report.
[17]. P.Warstein, H.Situ and Z.Huang(2010), "Load balancing in a cluster computer" In proceeding of the seventh International Conference on Parallel and Distributed Computing, Applications and Technologies, IEEE.

[18]. T. Kokilavani J.J. College of Engineering & Technology and Research Scholar, Bharathiar University, Tamil Nadu, India" Load Balanced Min-Min Algorithm for Static Meta-Task Scheduling in Grid Computing" International Journal of Computer Applications (0975 – 8887) Volume 20– No.2, April 2011.
[19]. Zenon Chaczko, Venkatesh Mahadevan, Shahrzad Aslanzadeh, Christopher Mcdermid (2011)"Availabity and Load Balancing in Cloud Computing" International Conference on Computer and Software Modeling IPCSIT vol.14 IACSIT Press,Singapore 2011.
[20]. Abhijeet Purohit, MA Waheed and Asma Parveen "Controlling Job Arrival and Processing in a Public Cloud".
[21]. www.images.google.com.