Cloud Management Systems - Load Balancing Algorithms and VDI Implementation Techniques

Micheal Ernest Taylor[1(✉)], David Aboagye-Darko[2], and Jian Shen[1]

[1] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, Jiangsu, People's Republic of China
delen007@live.com, s_shenjian@126.com
[2] Department of Information Technology, University of Professional Studies, Accra, Greater Accra, Ghana
aboagye.david@upsamail.edu.gh

Abstract. Internet technologies have upsurge the shift from earlier organizational computing processes which were characterized by main frames, client-server models and personal computers (PCs) and have restructured the concept of computing into a phenomenon that uses infrastructure across the globe. This has given rise to IT's reliance on heterogeneous network services and corresponding protocols. Internet technologies and virtualization are key features of cloud management systems. Cloud management system, considered as an evolutionary development, has made the provision of cloud services possible. This is premised on knowledge advancement in technological innovation regarding virtualization, automation of data-centers and network connectivity. A cloud-based VDI solution is a computing model where an end user's system can access all the essential files and data virtually in spite of being alienated from the physical IT infrastructure. However, issues of load balancing and VDI implementation undermine the Quality of Experience (QoE) users obtain from cloud management systems. This study adopts qualitative metrical analysis and comparative metrics on load balancing algorithms to determine algorithms that ensure workload balance. This paper focuses on load balancing algorithms in cloud management systems and virtual desktop infrastructure. The paper further presents rules for implementing a load balancer in cloud management systems, techniques for VDI implementation and a corresponding matrix table.

Keywords: Load balancing · Virtualization · Algorithms · Cloud management systems

1 Introduction

Cloud management system, which is considered as an evolutionary development, has made the provision of cloud services possible [2]. This is premised on knowledge advancement in technological innovation regarding virtualization, automation of data-centers and network connectivity [6]. Cloud management systems are considered as a

X. Sun et al. (Eds.): ICAIS 2019, LNCS 11633, pp. 561–570, 2019.
https://doi.org/10.1007/978-3-030-24265-7_48

relatively new IT-service innovation which has received attention in research and practice. This is in response to the emergence and steady dominance of the service economy; there is a high demand for services in various forms which has led to the addition of service dimensions to most products, including technology [1]. Internet technologies and virtualization are key features of cloud management systems. According to [4], internet technologies have given rise to a shift from earlier organizational computing processes which were characterized by main frames, client-server models and personal computers (PCs) and have restructured the concept of computing into a phenomenon that uses infrastructure across the globe. This has given rise to IT's reliance on heterogeneous network services and corresponding protocols. A cloud-based virtual desktop infrastructure (VDI) solution is a computing model where end users' system can access all the essential files and data virtually in spite of being alienated from the physical IT infrastructure. The VDI layer acts as an intermediary between the backend and the end-user application. Organizations use VDI technologies, from vendors like Citrix and VMware, to manage virtual desktops across their enterprises [8]. A load balancer is a networking tool designed to distribute workload. An end user connects to the port side of the load balancer, while the modules of the application being scaled connect to the trunk side. When an end user's demand arrives, the load balancer directs it to a component based on a fair scheduling algorithm. Public cloud providers, such as AWS, Google and Microsoft Azure, offer load balancing tools on their platforms. In this paper, a discussion on Load balancing algorithms, strategies for implementation in Cloud Systems and Virtual Desktop Infrastructure are presented. The rest of the paper is organized as follows; (1) the second section presents related works, (2) the subsequent section describes in detail the architecture of Cloud Management System and Virtual Desktop Infrastructure (VDI), (3) a load balancing algorithms comparison table, techniques for VDI implementation and a corresponding matrix table are also presented. This paper concludes with remarks on the state of Cloud management systems and virtual desktop infrastructure and its modules (Figs. 1 and 2).

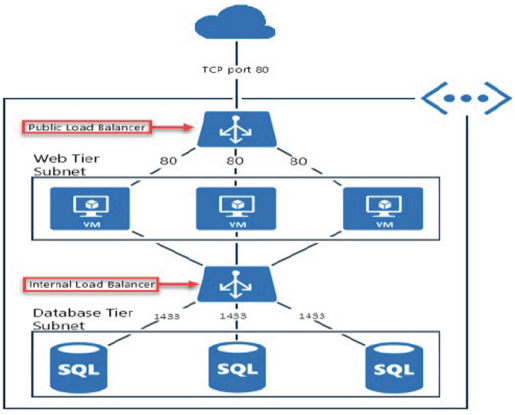

Fig. 1. Load balancing multi-tier applications by using both public and internal load balancer. Source: https://docs.microsoft.com.

2 Problem Statement

Load arrives randomly in a cloud computing systems and can clog the server's bandwidth, overloading some nodes while others are idle. This problem persists and propagates with the cumulative customer base in the cloud system. The solution to this problem is to ensure equal distribution of work load amongst all the nodes in the cloud by efficient planning and resource distribution, refining the overall performance of the system, reducing the response time and total cost of the system. A necessary qualitative metrical analysis and the comparison of various load balancing algorithms is required to achieve Quality of Experience (QoE).

3 Related Works

A number of studies have been conducted to understand Cloud Computing systems, load balancing and virtualization, it's favorable as well as adverse effects on both the CSPs and consumers. Kohler [3] suggests that virtualization emerged as a result of progress made regarding network interconnectivity, bandwidth and three (3) dimensional graphics. Virtualization involves technology generated platforms, physical locations or situations that are generated graphically [3]. Virtualization facilitates real-time, rich media content and great collaboration between individuals and organizations. Internet technologies and virtualization are essential features of cloud services that enhances its value proposition to customers. However, one of the major lapses that undermines the quality of service and performance in cloud computing is load balancing. Load balancing is a technique that facilitates the provision of effective resource time and utilization by decentralizing the overall load to several cloud nodes [5]. Load balancing in cloud management systems engenders optimum performance and effective utilization of resources. Authors in [7] suggest that vendors of cloud management systems consistently engage in proliferation in the provision of services to organizations, hence the load imbalance that characterizes dynamic distribution of services to clients or subscribers. The writers suggest that the issue of load imbalance can be addressed by adopting load balancing algorithms that ensures a balance in workload. This study represents an attempt to address the issue of load balancing in cloud management systems by combining load balancing algorithms that ensure workload balance.

4 Contribution

4.1 Load Balancing Challenges, Strategies and Rules of Implementation

Load balancing in cloud systems is major challenge. As such, we suggest these strategies and rules for successful implementation.

- An effective load balancer must connect to end users and to its scaled application modules: This can be achieved by adding or removing application modules, similarly adding and withdrawing the load balancer's trunk ports. If a module fails and is replaced, an update of the trunk port address of that module is needed. This is problematic for many enterprises, as load balancers are often part of network middleware.

- Performance Management and Quality of Experience: It is doubtful to have same network connections and performance in using public cloud environments as in your data center and most often they are not even similar. Various enterprises including those that use the open internet, have relatively slow connections to CSPs. Network latency and performance can vary, which means that new component instances will perform differently depending on whether they are in the cloud or data center. This variability can be a problem and also confound capacity planning.
- Management of State Control: Most industry applications are transactional, meaning they encompass various messages within a given dialog amid a user and an application. With load balancing, messages related to a particular transaction could be sent to different modules. If those modules are stateful, that is they expect to process transactions as a whole, the result can be a software failure or a corrupted database.

4.2 Rules of Implementation

In order to address load balancing challenges associated with using cloud systems in an organization, the IT teams need in-depth knowledge and critical planning. The following strategies can be implemented:

1. Front-end processing in the cloud is necessary when designing applications: Many enterprises use this method, nonetheless it is not widespread. When implementing front-end processing in the cloud, it is important to use the cloud scalability and load balancing services in the sections where they mostly required, such as the point of end-user connection. Also, this model enables multi-message transactions to trust into a single message which eradicates the issue of state control.
2. Design Load balancer for accessibility and availability: In a hybrid cloud system with a cloud frontend, the load balancer must be placed in the cloud, load balance data center modules and place the load balancer in the data center. This allows an easy update of load balancers to maintain a connection with all the components it supports. Load balancer availability is critical and often overlooked. Most CSPs design their own load balancers for high availability. In a situation where an organization implements its own load balancers, connection to a new instance of the load balancer must be supplied incase the old instance fails. Virtualization, in any form, stimulates scalability. As container orchestrators, such as Kubernetes and other hosting tools progress, we expect to see more load balancing options as well as more risk that they will all work properly in a hybrid cloud architecture.
3. An implementation of a policy-based scalability in the organizations hybrid cloud architecture: A given module should generally scale within its inherent hosting setting whenever possible and scale between the cloud and the data center in case of failure or lack of resources. Furthermore, plan the capacity of connection to public cloud services carefully to handle any workflows that have to cross between the

cloud and data center by limiting the number of cases where the public cloud and data center back each other up. This helps to assess the connectivity requirements between the cloud systems and ensures that the data center recovery strategy does not just create a cloud connection problem. Also, scale modules within restricted resource pools (such as a single data center) and closely connected data centers or a single cloud provider. This method will possibly progress performance steadiness and make it easier to update the load balancer with the addresses of new modules.

5 Load Balancing Algorithms Comparison

Load Balancing algorithms are used to improve the overall performance of the cloud systems. Cost, scalability, flexibility and its executing flow are some major factors that decide the effectiveness and efficiency of an algorithm. In cloud computing, different load balancing algorithm have been proposed which the main tenacity is to realize high throughput and least response time. Basically, load-balancing algorithms are categorized into two: Static load balancing algorithm and Dynamic load balancing algorithm. The successive load balancing parameters are presently dominant in clouds.

- Throughput – It is the amount of work all nodes can process in a specific time period.
- Response time – The elapsed time between the demand placed and the beginning of a response after completion of the job.
- Scalability – Ability of a computer application (hardware/software/service) to continue its function effectively even when its size and topography changes.
- Priority: Preference of tasks based on factors like cost, time and size [9]
- Fault tolerance: A system designed such that it can tolerate and continue functioning amidst any failure.
- Overhead: Refers to the processing time required by the system for installation, operation or any transaction.
- Power Consumption – Information technology consumes tremendous power and involves high energy costs. Efficient power management is vital for the success of IT environment such as cloud computing systems.
- Complexity – Making the entire system difficult. With increasing users associating with the cloud and its properties, the complexity of the system increases.
- Fairness – Indicates that each user has the equal response time and all get their jobs completed within approximately the same time.
- Performance – It is the speed and accuracy at which the jobs are completed and is measured against the preset standards. In simple terms, it is the total efficiency of the system. This can be improved by reducing the task response time and waiting time maintaining a reasonable cost of the system (Table 1).

Table 1. Load balancing algorithms metric table

Algorithms/parameters	ESCE	Round robin	Throttled	Ant colony	Task scheduling
Dynamic/Static	Dynamic	Static	Dynamic	Dynamic	Dynamic
Throughput	Average	Low	Average	High	Good
Response Time	Average	Low	Average	Good	Good
Scalability	Average	Low	High	High	High
Priority	High	Low	High	High	High
Fault tolerance	Low	Low	Average	High	High
Overhead	Average	High	Average	Low	Low
Power consumption	High	High	High	Average	Average
Complexity	Low	Low	Low	Average	High
Fairness	Average	Low	Average	High	Average
Performance	Average	Low	Average	Average	High

Percentage Metrics: Low = 0–39%: Average = 40–69%: High = 70–99%.

Load Balancing affects cloud systems and improves its performance by redistributing the load among the processors with jobs transferred from one node to another through the network involving some delay (queuing delay + processing delay) as it has to determine the destination node through remote processing. In [10], a distributed system model has n no. of users and m no. of computing resources. Nash Equilibrium can be defined for a distributed system model as a strategy "s" for every user "u" as

$$\mathrm{Su} = \mathrm{ArgMin\ Dj\ (S1,\ S2.\ldots Su.\ldots Sn)} \tag{1}$$

The Nash equilibrium is realized when no user can decrease its average expected response time by individually changing its strategy. A cloud system which has a normal rate should dispatch jobs instantaneously when it receives from the nodes that would process them. These processors maintain a waiting queue, given the following equations

$$\mathrm{TRT} = \mathrm{Pt} + \mathrm{Wt} + \mathrm{Tt} \tag{2}$$

Where TRT = Total Response Time
Pt = Processing time, Wt = Waiting time, Tt = Transfer time.

$$\mathrm{RPR} = \mathrm{JPR} \ / \ \mathrm{LPR\ in\ cluster} \tag{3}$$

Where RPR = Response Processing Rate, JPR = Jobs processing Rate, LPR = Lowest Processing Rate.

$$\mathrm{JGR} = \mathrm{UJGR/TJGR\ of\ all\ users} \tag{4}$$

Where JGR = Job Generation Rate, UJGR = Users Job Generation Rate, TJGR = Total Job Generation Rate

6 Virtualization and Virtual Desktop Infrastructure

[11] Defines virtualization as "a means of abstracting a computer's physical resources into virtual ones with the help of specialized software. Abstraction layers allow for the creation of multiple VMs [virtual machines] on a single physical machine (ES. 5)".

Virtualization affords encapsulation of processing capabilities of IT resources into virtual platforms and executes these virtual platforms on a host machine in an isolated arena. Many organizations are exploiting a variety of virtual platforms in response to the challenges organizations are experiencing with regards to computing [12].

Presents two categories of virtualization namely hardware and operation system (OS) virtualization. It is important to note that the writers' taxonomy is premised on two fundamental principles. Firstly, the provision of a virtual hardware by the hypervisor which is identified as the system hardware. Secondly, the creation of containers that employ the host hardware through the host OS and the hypervisor. Application virtualization has been identified as a dominant emerging trend in virtual technologies. Thus, it may be considered as the third type of virtualization [12]. However, the focus of this study is on the infrastructure that undergirds desktop virtualization.

6.1 Virtual Desktop Infrastructure

VDI provides a framework for hosting a desktop operating system within a virtual machine (VM) on a server. A user at an endpoint workstation accesses the VM over the network via a remote display protocol that allows the virtualized desktop to be rendered locally. Desktop virtualization is the concept of isolating a logical operating system (OS) instance from the client that is used to access it. There are several different conceptual models of desktop virtualization, which can broadly be divided into two categories based on whether or not the operating system instance is executed locally or remotely. It is important to note that not all forms of desktop virtualization involve the use of virtual machines (VMs). Virtual desktop infrastructure (VDI) is the practice of hosting a desktop operating system within a virtual machine (VM) running on a centralized server [11]. VDI is a variation on the client/server computing model, sometimes referred to as server-based computing. The term was coined by VMware Inc. Many organizations are turning to VDI as an alternative to the server-based computing model used by Citrix and Microsoft Terminal Services.

7 Techniques for Implementing a VDI in Cloud Systems

Successful implementation is necessary in order to exploit the potential benefits (such as centralized IT management) of VDI in cloud systems. The implementation of VDI in cloud systems is dependent not only on the technical reality of IT [13], but also on the complex dynamics that characterize the process by focusing on the needs of users. Therefore, it is imperative that in understanding the techniques for VDI implementation in cloud systems, we also examine the effect of VDI deployment.

User location: When implementing VDI in cloud systems, it is important to consider the location of users. A lot of bandwidth is required for the workstation to work effectively when the distance between the user and data center is greater. IT can address the desktop needs of remote and mobile users by delivering desktops to specific off-site locations. This can be accomplished by upgrading network hardware components (e.g. WAN link controllers to mitigate issues of bottlenecking). When the distance between the user and the data center is great, the cost involved in delivering remote desktops to users will be high.

User Behavior: The behavior of system users such as the kind of applications they use, when they start their day and when they boot their devices influence the VDI needs of an organization. The addition of input output operation per second (IOPS) improves the performance of VDI during peak usage times.

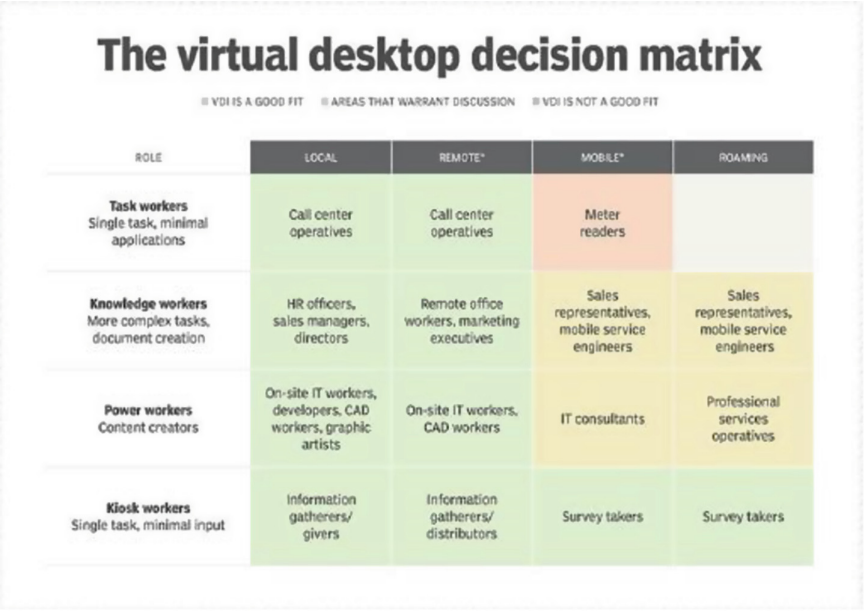

Fig. 2. VDI decision matrix Source: www.techtarget.com

Knowledge, task and kiosk workers can use non-persistent desktops and more memory. Power workers require persistent desktops and high memory requirements in view of the fact that they use a variety of applications that consume a lot of resources.

7.1 Effects of VDI on Other Systems

Issues of bandwidth consumption and IP address confront systems after VDI implementation. On the one hand, one IP address is required for the desktop and device in organizations that deploy physical desktops with thick clients. On the other hand,

organizations that deploy thin clients require two IP addresses, one for the thin client itself and the other for the virtual desktop. In view of the aforementioned, it is imperative that organizations adopt IP address management tools to address issues of bandwidth consumption and anticipation of required IP addresses per number of users.

8 Conclusion

In this paper, the rules for implementing a load balancer in cloud management systems as well as the techniques for the implementation of VDI were presented. The potential benefits cloud management systems present have resulted in its implementation in several organizations across the globe. Unfortunately, the issue of load balancing and VDI implementation are some of the major lapses that undermine the QoE users obtain while using cloud management systems [14]. To this end, the qualitative metrical analysis and comparative metrics were conducted on load balancing algorithms in VDI and cloud management systems to determine algorithms that ensure workload balance. It was argued that in order to implement load balancers in cloud management systems, rules such as front-end processing in the cloud when designing applications, designing the load balancer for accessibility and availability, and the implementation of a policy based scalability in hybrid cloud architecture are necessary for optimum performance of cloud management systems. Also, user location, user behavior as well as the effect of VDI on other systems should be considered during VDI implementation in cloud systems. Further studies can be conducted to identify combinations of specific load balancing algorithms that completely addresses issues such as load balancing ports and trunks, as well as state control, in cloud management systems.

References

1. Fang, E., Palmatier, R.W., Steenkamp, J.: Effect of service transition strategies on firm value. J. Mark. **72**(5), 1–14 (2008)
2. Iyer, B., Henderson, J.C.: Preparing for the future by understanding the seven capabilities cloud computing. MIS Q. **9**(2), 117–131 (2010)
3. Kohler, T.: Co-creation in virtual worlds: the design of the user experience. MIS Q. **9**(2), 773–788 (2003)
4. Lyytinen, K., Rose, G.M.: The disruptive nature of information technology innovations: the case of internet computing in systems development organizations. MIS Q. **27**(4), 557–596 (2003)
5. Nakai, A.M., Madeira, E., Buzato, L.E..: Improving the QOS of web services via client-based load distribution. In: 29th Proceedings of Brazilian Symposium on Computer Networks and Distributed Systems, pp. 617–629. Unicamp (2011)
6. Venters, W., Whitley, E.A.: A critical review of cloud computing: researching desires and realities. J. Inf. Technol. **27**(1), 179–197 (2012)
7. Taylor, M.E., Shen, J.: Cloud management systems and virtual desktop infrastructure load balancing algorithms - a survey. In: Sun, X., Chao, H.-C., You, X., Bertino, E. (eds.) ICCCS 2017. LNCS, vol. 10602, pp. 300–309. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68505-2_26

8. Liu, J., Lai, W.: Security analysis of VLAN-based virtual desktop infrastructure. In: International Conference on Educational and Network Technology, pp. 301–304 (2010)
9. Gouda, K.C., Radhika, T.V., Akshatha, M.: Priority based resource allocation model for cloud computing. Int. J. Sci., Eng. Technol. Res. **2**(1), 215–219 (2013)
10. Taylor, M.E., Shen, J.: Cloud management systems and virtual desktop infrastructure load balancing algorithms - a survey. In: Sun, X., Chao, H.C., You, X., Bertino, E. (eds.) Cloud Computing and Security. ICCCS 2017, LNCS, vol. 10602, pp. 1–13. Springer, Cham (2017)
11. Chronopoulos, A.T.: Game Theory Based Load Balanced Job Allocation. http://graal.enslyon.fr/∼lmarchal/aussois/slides/chronopoulos.pdf. Accessed 04 Dec 2018
12. Lunsford, D.L.: Virtualization technologies in information systems education. J. Inf. Syst. Educ. **20**(3), 339–348 (2017)
13. Xie, X., Yuan, T., Zhou, X., Cheng, X.: Research on trust model in container-based cloud service. CMC **56**(2), 273–283 (2018)
14. Cheang, C.F., Wang, Y., Cai, Z., Xu, G.: Multi-Vms intrusion detection for cloud security using dempster-shafer theory. CMC **57**(2), 297–306 (2018)