Proceedings of the First PhD Symposium on Sustainable Ultrascale Computing Systems (NESUS PhD 2016)
Timisoara, Romania

Jesus Carretero, Javier Garcia Blas
Dana Petcu
(Editors)

February 8-11, 2016

Cloud Resource Management

Tychalas Dimitrios
PhD Student
Aristotle University of Thessaloniki, Greece
dtychala@csd.auth.gr

Karatza Helen
Supervisor
Aristotle University of Thessaloniki, Greece
karatza@csd.auth.gr

Abstract

Nowadays computational needs increase exponentially every year. We analyze, calculate and process large data sets every day and the "traditional" servers do not meet these computational criteria. As a result cloud computing was "invented" offering multiple resources at an affordable cost. Besides that, Cloud Computing supports scalability, fault tolerance and high availability [2] [16]. Our goal is to delve deeper into Cloud Computing to be able to carry out independent research to study and improve the state of the art load balancing techniques.

Keywords Ultrascale systems, NESUS, Cloud computing, Load balancing, Fault tolerance, High availability, Scalability

I. Introduction

Cloud computing is one of the most fast-growing fields in computer science [2]. Almost everyone has access to Internet via his smart-phone/tablet/PC [18] and access his data from anywhere. In the near future everything would be on the "cloud" making the network needs to grow exponentially. As a result the next-generation of cloud computing will thrive on how effectively the infrastructure is used and if the available resources can be utilized dynamically [1]. Load balancing distributes the load across multiple virtual machines to ensure that the service is always accessible and the resources are utilized in the best effort. Moreover a "good" load balancer should adapt its decisions to the changing environment [17] [19].

The main goal of this thesis is to examine the known load balancing techniques and algorithms and improve them in the cost and energy saving aspects [19].

II. Related work

The most used load balancing techniques [15] are:

1. **Round Robin:** Incoming requests are distributed sequentially across the available virtual machines. All virtual machines should be homogeneous.
2. **Weighted Round Robin:** Incoming requests are distributed across the virtual machines in a sequential manner, while taking account of a static "weight" that can be pre-assigned per VM. This method is preferred on heterogeneous VMs.
3. **Least Connection:** Incoming requests are distributed on the basis of the connections that every VM is currently maintaining. The VM with the least number of active connections automatically is selected.
4. **Weighted Least Connection:** Incoming requests are distributed across the virtual machines with the fewer active connections, while taking account of predefined "weight" for each VM.

There are a number of works that are employing load balancing algorithms that take in account current requirements for CPU performance like [4] [8] [9] [20]. However despite the high performance achieved by the aforementioned algorithms, they lead to high energy consumption. This resulted in the development of many routing algorithms for power awareness as [11] [21] [24].

III. Thesis idea

Cloud computing is so involved in our every day lives and spread among many different aspects of research. It is the ideal area for aspiring computer scientists to keep themselves up to date with the latest technologies. In our research we will study the load balancing technologies and we will address open issues.

In order to examine the state of the art algorithms and techniques in this field, we first developed a Web Framework that uses more than one Virtual Machine in order to address the problems of the "classic" servers. The main problems are faults, as power failure, errors on system or on hardware, expensive hardware when scalability is needed and of course the overloading on the server when multiple users are connected simultaneously. The system is intended to deal with all the aforementioned problems using:

1. Virtual Machines, by ∼okeanos [12]
2. MySQL Cluster [3] [13]
3. Apache as Load Balancer [6] [14]
4. GlusterFS [23]

The system employs load balancing to handle the multiple requests. There are many ways to balance traffic between systems [15], but the most effective one is using weights. The weight is determined by counting the requests that each server has and how much time is needed to serve all of them. The output of this study was published in [22].

Secondly we utilized the package JPPF (Java Parallel Processing Framework) which enables applications with large processing power requirements to be run on any number of computers. This is done by splitting an application into smaller parts and executes them simultaneously on different machines [7]. We used the above package in order to write our own load balancing rules and use it in co-operation between a Desktop PC and a Raspberry. Our load balancing algorithm works with meta-tags in every task. If the meta-tags of a task meet the minimum needs, then the Raspberry is used in order to process the task, alternatively the task is processed by the Desktop.

Finally we are developing our own simulation program in C in order to test the above systems with more virtual machines or with more Servers - Raspberries.

IV. Future work

As future work we are going to use KVM [5] as virtualization solution because we can increase or decrease the number of CPUs and the amount of RAM on-the-fly, without the need of restarting the virtual machine [10]. As a result we can increase the resources when it is needed and decrease them in order to save energy and money.

V. Acknowledgment

We would like to acknowledge the contribution of the academic cloud service ∼okeanos [12] for giving us the ability to create the necessary virtual machines for the above case study. We would also like to acknowledge the contribution of the COST Action IC1305 NESUS (Network for Sustainable Ultrascale Computing).

References

[1] Omer F. Rana Antonio Corradi. “The management of cloud systems”. In: *Future Generation Computer Systems* 32 (2014), pp. 24–26.

[2] Michael Armbrust et al. “A view of cloud computing”. In: *Communications of the ACM* 53.4 (2010), pp. 50–58.

[3] Charles Bell, Mats Kindahl, and Lars Thalmann. *MySQL high availability: tools for building robust data centers*. " O’Reilly Media, Inc.", 2010.

[4] Anton Beloglazov and Rajkumar Buyya. “Energy Efficient Resource Management in Virtualized Cloud Data Centers”. In: *Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*. CCGRID ’10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 826–831. ISBN: 978-0-7695-4039-9. DOI: 10.1109/CCGRID.2010.46. URL: http://dx.doi.org/10.1109/CCGRID.2010.46.

[5] Anton Beloglazov et al. "Deploying OpenStack on CentOS using the KVM Hypervisor and GlusterFS distributed file system". In: *Cloud Computing and Distributed Systems (CLOUDS) Laboratory Department of Computing and Information Systems, The University of Melbourne, Australia* (2012).

[6] Trieu C Chieu et al. "Dynamic scaling of web applications in a virtualized cloud computing environment". In: *e-Business Engineering, 2009. ICEBE'09. IEEE International Conference on*. IEEE. 2009, pp. 281–286.

[7] L. Cohen. *Java Parallel Programing Framework*. 2005. URL: http://www.jppf.org (visited on 12/26/2015).

[8] Shridhar G Domanal and G Ram Mohana Reddy. "Optimal load balancing in cloud computing by efficient utilization of virtual machines". In: *Communication Systems and Networks (COMSNETS), 2014 Sixth International Conference on*. IEEE. 2014, pp. 1–4.

[9] James Michael Ferris. *Load balancing in cloud-based networks*. US Patent 8,849,971. Sept. 2014.

[10] *Hotplug (qemu disk,nic,cpu,memory)*. 2015. URL: https://pve.proxmox.com/wiki/Hotplug_(qemu_disk,nic,cpu,memory) (visited on 12/26/2015).

[11] Myungsun Kim et al. "Utilization-aware load balancing for the energy efficient operation of the big. LITTLE processor". In: *Proceedings of the conference on Design, Automation & Test in Europe*. European Design and Automation Association. 2014, p. 223.

[12] Vangelis Koukis, Constantinos Venetsanopoulos, and Nectarios Koziris. "~ okeanos: Building a Cloud, Cluster by Cluster". In: *IEEE Internet Computing* 3 (2013), pp. 67–71.

[13] Arjen Lentz. "MySQL Cluster Introduction". In: *White Paper* (2006).

[14] Quanzhong Li and Bongki Moon. "Distributed cooperative Apache web server". In: *Proceedings of the 10th international conference on World Wide Web*. ACM. 2001, pp. 555–564.

[15] *Load Balancing Scheduling Methods Explained | LoadBalancerBlog.com*. 2013. URL: http://loadbalancerblog.com/blog/2013/06/load-balancing-scheduling-methods-explained (visited on 12/26/2015).

[16] Ioannis A Moschakis and Helen D Karatza. "Enterprise HPC on the Clouds". In: *Cloud Computing for Enterprise Architectures*. Springer, 2011, pp. 227–246.

[17] Ioannis A Moschakis and Helen D Karatza. "Evaluation of gang scheduling performance and cost in a cloud computing system". In: *The Journal of Supercomputing* 59.2 (2012), pp. 975–992.

[18] Ioannis A Moschakis and Helen D Karatza. "Towards scheduling for Internet-of-Things applications on clouds: a simulated annealing approach". In: *Concurrency and Computation: Practice and Experience* (2013).

[19] Ioannis Moschakis, Helen D Karatza, et al. "Performance and cost evaluation of Gang Scheduling in a Cloud Computing system with job migrations and starvation handling". In: *Computers and Communications (ISCC), 2011 IEEE Symposium on*. IEEE. 2011, pp. 418–423.

[20] Kumar Nishant et al. "Load balancing of nodes in cloud using ant colony optimization". In: *Computer Modelling and Simulation (UKSim), 2012 UKSim 14th International Conference on*. IEEE. 2012, pp. 3–8.

[21] George Terzopoulos and Helen Karatza. "Power-aware load balancing in heterogeneous clusters". In: *Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2013 International Symposium on*. IEEE. 2013, pp. 148–154.

[22] Dimitris Tychalas and Helen Karatza. "A cloud system for health care". In: *Proceedings of the 19th Panhellenic Conference on Informatics*. ACM. 2015, pp. 169–170.

[23] YANG Yong. "Distribution Redundancy Storage Based on GlusterFS". In: *Journal of Xifffdfffdfffdan University of Arts & Science (Natural Science Edition)* 4 (2010), pp. 67–70.

[24] Andrew J Younge et al. “Efficient resource management for cloud computing environments”. In: *Green Computing Conference, 2010 International*. IEEE. 2010, pp. 357–364.