ISSN 2320-9984

Volume 3 Issue 1 | March 2015

ISSN: 2320-9984 (Online)

International Journal of Modern Engineering & Management Research

Website: www.ijmemr.org

An Approach to Maximize Load Balancing in Perspective of Cloud Database Environment Studies and Consequences

Kalpana Mahant
Research Scholar
Lakshmi Narain College of Technology
Jabalpur, (M.P.), [INDIA]
Email : kalpana.mahant@gmail.com

Prof. Sujeet Kumar Tiwari
Professor & Head
Department of Computer Science and Engineering
Lakshmi Narain College of Technology
Jabalpur, (M.P.), [INDIA]
Email : sujeet.tiwari08@gmail.com

Abstract—*Creating a high-availability server environment is an important step to scale, improve performance and maintain application stability on Joyent Cloud. Load balancers can help orchestrate traffic to your websites, perform health checks and proactively remove a node from rotation to ensure maximum availability. This manuscript presents an exhaustive survey on cloud computing technology and attempts to cover most of the developments that have taken place in the field of cloud computing. It discusses about the various available cloud computing platforms, Security in cloud, reference architectures for cloud and storage of data in cloud computing.*

1. INTRODUCTION

Cloud computing is a technology for the future and will change the entire scenario of the IT industry, being a cost efficient approach, with reduced exigency of buying the software or the hardware resources. It is an on demand form of utility computing for those who have access to cloud. [1-3]

Even though the cloud has greatly simplified the capacity provisioning process, it poses several novel challenges in the area of Quality-of-Service (QoS) management. QoS denotes the levels of performance, reliability, and availability offered by an application and by the platform or infrastructure that hosts it. QoS is fundamental for cloud users, who expect providers to deliver the advertised quality characteristics, and for cloud providers, who need to find the right tradeoffs between QoS levels and operational costs. However, finding optimal tradeoff is a difficult decision problem, often exacerbated by the presence of service level agreements (SLAs) specifying QoS targets and economical penalties associated to SLA violations [3]. Recent web search trends have shown a paradigm shift in peoples interest towards cloud. As per Google search trends [4] there has been an immense increase in people's interest towards cloud computing from 2005 to 2013, also shown by Figure 1.

The commercialization of these developments is defined currently as Cloud computing [2], where computing is delivered as utility on a pay-as-you-go basis. Traditionally, business organizations used to invest huge amount of capital and time in acquisition and maintenance of computational resources. The emergence of Cloud computing is rapidly changing this ownership-based approach to subscription-oriented approach by providing access to scalable infrastructure and services on-demand. Users can store, access, and share any amount of information in Cloud. That is, small or medium enterprises/ organizations do not have to worry about purchasing, configuring, administering, and maintaining their own computing infrastructure [5, 6]

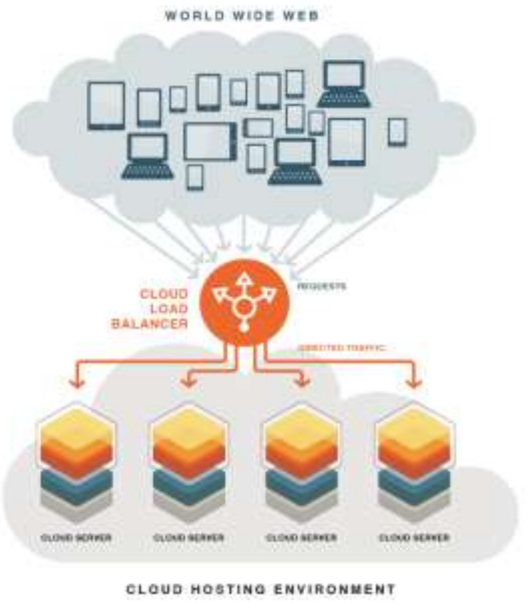


Figure 1 Cloud Hosting Environment

Azure Traffic Manager allows you to control the distribution of user traffic to endpoints, which can include cloud services, websites, external sites, and other Traffic Manager profiles. Traffic Manager works by applying an intelligent policy engine to Domain Name System (DNS) queries for the domain names of your Internet resources. Your cloud services or websites can be running in different datacenters across the world. [7-9]

2. TRAFFIC MANAGER LOAD BALANCING FOR CLOUD SERVICES AND WEBSITES

Failover: Use this method when you want to use a primary endpoint for all traffic, but provide backups in case the primary becomes unavailable.

Performance: Use this method when you have endpoints in different geographic locations and you want requesting clients to use the "closest" endpoint in terms of the lowest latency.

Round Robin: Use this method when you want to distribute load across a set of cloud services in the same datacenter or across cloud services or websites in different datacenters. [10-12]

Infrastructure-user load balancing

Hybrid clouds are considered in [13,14]. The authors formulate an optimization problem faced by a cloud procurement endpoint (a module responsible for provisioning resources from public cloud providers), where heavy workloads are tackled by relying on public clouds. They present a linear integer program to minimize the resource cost, and evaluate how the solution scales with the different problem parameters. Finally,[15] proposes an adaptive approach for component replication of cloud applications, aiming at finding a cost-effective placement and load balancing. This is a distributed method based on an economic multi-agent model that achieves high application availability guaranteeing at the same time service availability under failures.

Azure load balancing for virtual machines

Virtual machines in the same cloud service or virtual network can communicate with each other directly using their private IP addresses. Computers and services outside the cloud service or virtual network can only communicate with virtual machines in a cloud service or virtual network with a configured endpoint. An endpoint is a mapping of a public IP address and port to that private IP address and port of a virtual machine or web role within an Azure cloud service.

The Azure Load Balancer randomly distributes a specific type of incoming traffic across multiple virtual machines or services in a configuration known as a load-balanced set. For example, you can spread the load of web request traffic across multiple web servers or web roles. [16-18]

Green Cloud Architecture

From the above study of current efforts in making Cloud computing energy efficient, it shows that even though researchers have made various components of Cloud efficient in terms of power and performance, still they lack a unified picture. Most of efforts for sustainability of Cloud computing have missed the network contribution. [19]

SaaS Level: Since SaaS providers mainly offer software installed on their own. Datacenters or resources from IaaS providers, the SaaS providers need to model and measure energy efficiency of their software design, implementation, and deployment. [20]

PaaS level: This can be done by inclusion of various energy profiling tools such as JouleSort. It is a software energy efficiency benchmark that measures the energy required to perform an external sort. In addition, platforms itself can be designed to have various code level optimizations which can cooperate with underlying complier in energy efficient execution of applications. [21]

IaaS level: Providers in this layer plays most crucial role in the success of whole Green Architecture since IaaS level not only offer independent infrastructure services but also support other services offered by Clouds. Various energy meters and sensors are installed to calculate the current energy efficiency of each IaaS providers and their sites. This information is advertised regularly by Cloud providers in Carbon Emission Directory. [22]

3. MULTITENANCY MODELS

Multitenancy i.e. resource sharing amongst different tenants, is important to serve applications that have small but varying resource requirements [14, 21, 23]. SaaS providers like Salesforce.com [21] are the most common examples of multitenancy in both the application as well as the database tier. A tenant is an application’s database instance with its own set of clients and data. Different multitenancy models arise from resource sharing at different levels of abstraction; the shared machine, shared process, and shared table models are well known [14].

Experimental Setup

Cluster Configuration Experiments were performed on a six node cluster, each with 4 GB memory, a quad core processor, and a 250 GB disk. The distributed fault-tolerant storage and the OTMs are co-located in the cluster of five worker nodes. The TM master (controller) and the clients generating the workloads were executed on a separate node Figure 2. [23] Each OTM was serving 20 tenants on average. In all experiments, except the one presented in Appendix B.3.4, we evaluate migration cost when both NSRC and NDST were lightly loaded, so that the actual overhead of migration can be measured. The load on a node is measured using the amount of resources (for instance CPU cycles, disk I/O bandwidth, or network bandwidth) being utilized at the node. When resource utilization is less than 50%, it is referred to as lightly loaded, utilization between 15 – 75% is referred to as moderately loaded, and utilization above 70% is called overloaded. We only consider CPU utilization. [24]

Benchmarks*:* We evaluate migration cost using two OLTP benchmarks: the Yahoo! cloud serving benchmark (YCSB) and the TPC -C benchmark. YCSB is a recently proposed benchmark to evaluate systems that drive web applications. The initial benchmark was designed for Key-Value stores and hence did not support transactions. [25]

[25] H. Liu, H. Jin, X. Liao, L. Hu, and C. Yu. Live migration of virtual machine based on full system trace and replay. In HPDC, pages 101–110, 2009.

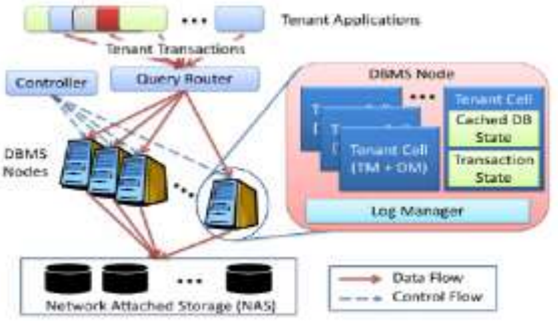
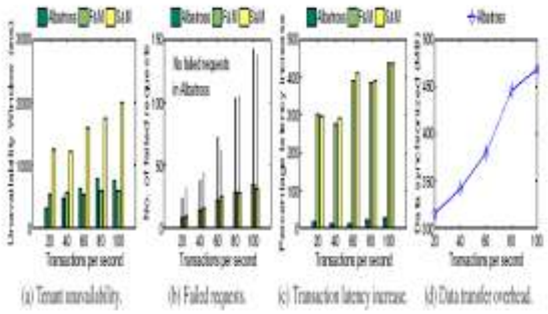


Figure 2: Experimental Setup



(a) Tenant unavailability. (b) Failed requests. (c) Transaction latency increase. (d) Data transfer overhead.

Evaluating migration cost using YCSB by varying the transaction load on the tenant. For failed requests , the wider bars represent aborted transactions and narrower bars represent failed operations.

Figure 3: Evaluating Migration cost (a, b, c, d)

Cloud Security Risks:

The security risks associated with each cloud delivery model vary and are dependent on a wide range of factors including the sensitivity of information assets, cloud architectures and security control involved in a particular cloud environment[26] We discuss these risks in a general context, except where a specific reference to the cloud delivery model is made these are risk that will be discussed in our forthcoming publications. [27]

4. CONCLUSION

Cloud-based projects can be conceived, developed, and tested with smaller initial investments than traditional IT investments. Rather than laboriously building data center capacity to support a new development environment, capacity can be provisioned in small increments through cloud computing technologies. After the small initial investment is made, the project can be evaluated for additional investment or cancellation. Projects that show promise can gain valuable insights through the evaluation process. Less promising projects can be cancelled with minimal losses. This “start small” approach collectively reduces the risk associated with new application development. Reducing the minimum required investment size will also provide a more experimental development environment in which innovation can flourish.

REFERENCES :

[1] Armbrust M, Fox A, Griffith R, Joseph AD, Katz R, Konwinski A, Lee G, Patterson D, Rabkin A, Stoica I, Zaharia M (2010) A view of cloud computing. Commun ACM 53(4):50-58Publisher Full Text

[2] Zhang Q, Cheng L, Boutaba R (2010) Cloud computing: state-of-the-art and research challenges. J Internet Serv Appl 1(1):7-18 Publisher Full Text

[3] Rdagna D, Panicucci B, Trubian M, Zhang L (2012) Energy-aware autonomic resource allocation in multitier virtualized environments. IEEE Trans Serv Comput 5(1):2-19

[4] Giri, Ravi A. 2010. Increasing Datacenter efficiency with server power measurements. http://download.intel.com/it/pdf/Server_Power_Measurement_final.pdf

[5] Allalouf, M., Arbitman, Y., Factor, M., Kat, R. I., Meth, K., and Naor, D. 2009. Storage modelling for power estimation. In Proceedings of 2009 Israeli Experimental Systems Conference (SYSTOR '09), Isreal.

[6] Allman, M., Christensen, K., Nordman, B., and Paxson, V. 2007. Enabling an EnergyEfficient Future Internet Through Selectively Connected End Systems, Proceedings of the Sixth ACM Workshop on Hot Topics in Networks (HotNets-VI), Atlanta, Georgia, USA.

[7] B.P. Rimal, C. Eunmi, I. Lumb, "A Taxonomy and Survey of Cloud Computing Systems," Fifth International Joint Conference on INC, IMS and IDC,pp.44,51, 25-27 Aug. 2009

[8] Google App Engine, https://

developers.google.com/appengine/, Jan 2014

[9] AbiCloud, http://www.abiquo.com/, accessed on: Jan 2014

[10] Jung G, Joshi KR, Hiltunen MA, Schlichting RD, Pu C (2008) Generating adaptation policies for multi-tier applications in consolidated server environments. In: Autonomic Computing, 2008 ICAC'08. International Conference on. IEEE, Chicago, IL, USA. pp 23-32

[11] Bacigalupo D, van Hemert J, Chen X, Usmani A, Chester A, He L Dillenberger D, Wills G, Gilbert L, Jarvis S (2011) Managing dynamic enterprise and urgent workloads on clouds using layered queuing and historical performance models. Simul Model Prac Theory 19:1479-1495

[12] Thereska E, Ganger GR (2008) IRONmodel: Robust performance models in the wild. ACM SIGMETRICS Perform Eval Rev 36 (1):253-264

[13] M.T. Khorshed, A.B.M.S. Ali, S. A. Wasimi, "A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing," Future Generation Computer Systems, vol. 28, no. 6, pp. 833-851, June 2012.

[14] D. Zissis, D. Lekkas, "Addressing cloud computing security issues," Future Generation Computer Systems, vol. 28, no. 3, pp. 583-592, March 2012.

[15] A. Chonka, Y. Xiang, W. Zhou, A. Bonti, Cloud security defence to protect cloud computing against HTTP-DoS and XML-DoS attacks, Journal of Network and Computer Application, vol. 34, no. 4, pp. 1097-1107, July 2011,

[16] Kusic D, Kephart JO, Hanson JE, Kandasamy N, Jiang G (2009) Power and performance management of virtualized computing environments via lookahead control. Cluster Comput 12(1):1-15

[17] Addis B, Ardagna D, Panicucci B, Squillante MS, Zhang L (2013) A hierarchical approach for the resource management of very large cloud platforms. IEEE Trans Dependable Secure Comput 10(5):253-272

[18] Goudarzi H, Pedram M (2011) Multi-dimensional sla-based resource allocation for multi-tier cloud computing systems. In: Proceedings of the 2011 IEEE 4th International Conference on Cloud Computing, CLOUD'11, 324–331, Washington, DC, USA

[19] Kephart, J. O., Chan, H., Das, R., Levine, D. W., Tesauro, G., Rawson, F., and Lefurgy, C. 2007. Coordinating multiple autonomic managers to achieve specified power-performance tradeoffs. Proceedings of 4th International Conference on Autonomic Computing, Florida, USA.

[20] Song, Y., Sun, Y., Wang, H., and Song, X. 2007. An adaptive resource flowing scheme amongst VMs in a VM-based utility computing. Proceedings of IEEE International Conference on Computer and Information Technology, Fukushima, Japan.

[21] Abdelsalam, H., Maly, K., Mukkamala, R., Zubair, M., and Kaminsky, D. 2009. Towards energy efficient change management in a Cloud computing environment, Proceedings of 3 rd International Conference on Autonomous Infrastructure, Management and

Security, The Netherlands.

[22] Kaushik, R. T., Cherkasova, L., Campbell, R., and Nahrstedt, K., 2010. Lightning: selfadaptive, energy -conserving, multi-zoned, commodity green Cloud storage system. In Proceedings of the 19th ACM International Symposium on High Performance Distributed computing (HPDC '10). ACM, New York, NY, USA.

[23] A. J. Elmore, S. Das, D. Agrawal, and A. El Abbadi. Zephyr: Live Migration in Shared Nothing Databases for Elastic Cloud Platforms. In SIGMOD, 2011.

[24] J. Gray. Notes on data base operating systems. In Operating Systems, An Advanced Course, pages 393–481, London, UK, 1978. Springer-Verlag.

[25] H. Liu, H. Jin, X. Liao, L. Hu, and C. Yu. Live migration of virtual machine based on full system trace and replay. In HPDC, pages 101–110, 2009.

[26] Vaquero, L. M., Rodero-Merino, L., Caceres, J., and Linder, M. (2009). A Break in the Clouds: Towards a Cloud Definition. ACM SIGCOMM Computer Communication Review, Vol 39, Issue 1, pp. 50-55, January 2009. Vouk, M. A. (2008).

[27] Cloud Computing – Issues, Research and Implementations. In Proceedings of the 30th International Conference on Information Technology Interfaces (ITI’08), pp. 31-40, Cavtat, Croatia, June 2008.