Response Time for Cloud Computing Providers

Mohammed Alhamad, Tharam Dillon, Chen Wu, Elizabeth Chang
Digital Ecosystems and Business Intelligence Institute (DEBII)
Curtin University of Technology
Perth, Australia
Mohammed.Alhamad@postgrad.curtin.edu.au,
Tharam.Dillon@cbs.curtin.edu.au, Chen.Wu@cbs.curtin.edu.au
Elizabeth.Chang@cbs.curtin.edu.au

Abstract. *Cloud services are becoming popular in terms of distributed technology because they allow cloud users to rent well-specified resources of computing, network, and storage infrastructure. Users pay for their use of services without needing to spend massive amounts for integration, maintenance, or management of the IT infrastructure. This creates the need for a reliable measurement methodology of the scalability for this type of new paradigm of services. In this paper, we develop performance metrics to measure and compare the scalability of the resources of virtualization on the cloud data centres. First, we discuss the need for a reliable method to compare the performance of cloud services among a number of various services being offered. Second, we develop a different type of metrics and propose a suitable methodology to measure the scalability using these types of metrics. We focus on the visualization resources such as CPU, storage disk, and network infrastructure. Finally, we compare well-known cloud providers using the proposed approach and conclude the recommendations. This type of research will help cloud consumers, before signing any official contract to use the desired services, to ascertain the ability and capacity of the cloud providers to deliver a particular service.*

Index Terms: Performance, SLA, Cloud computing, Trust management

1. Introduction

There have been various definitions proposed in the literature of cloud computing [1-3]. In this paper, we adopted and considered the definition provided by U.S. NIST (National Institute of Standards and Technology) that describes cloud computing as "... a model for enabling convenient, on demand network access to a share pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management afford or service provider interaction" [1]. In other words, as shown in Figure 1, cloud computing is a framework by means of which virtualized infrastructure resources are delivered as a service to customers by using a public network which is the Internet [4-6].

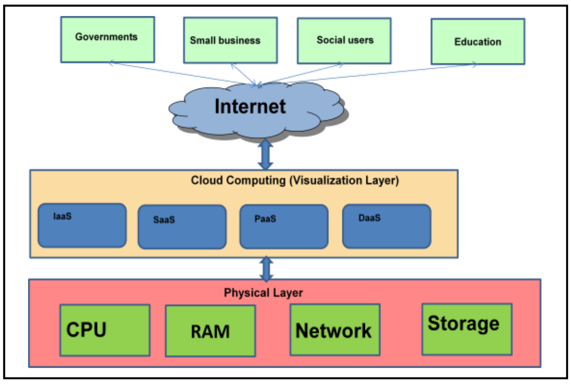

Figure 1. Cloud computing architecture

The cloud customers can range from big organizations, small business and developers to individual users. In this paper, we will refer to such customers as 'users'. One of the advantages of having such a framework is that users do not need to buy costly physical infrastructure or software, but they can use them over a virtual environment from other users at a much lower price, thereby reducing their operational and maintenance costs. For example, Salesforce.com developed a customer relationship management solution (CRM) and delivered this as a cloud service not as a package of software. Salesforce.com customers can use this type of service using a basic machine with an Internet browser [7]. There are four main delivery models of cloud services with such a paradigm. They are:

1. Infrastructure as a service (IaaS): In such architectures, users can use the visualization resources as a fundamental infrastructure for their applications. These resources may be a CPU, network, or storage. Cloud users can

ISBN: 978-1-4503-0421-4, iiWAS 2010. http://dx.doi.org/10.1145/1967486.1967579

manage the resources and assign rules for end users [8].

2. Database as a service (DaaS): Such architectures allow users to rent a specific size of storage for a specific period of time. Users are not required to manage the integration or the scaling of the infrastructure. Database providers take the responsibility for integration, privacy, and security of users' data [9].
3. Platform as a service (PaaS): In such architectures, users use all facilities on the cloud to develop and deliver their web application and services to the end users. PaaS services may include development, integration, testing or the storage resources to complete the life cycle of services [10].
4. Software as a service (SaaS): In such architectures, users connect with the service providers to use the application, but they do not control the infrastructure, operating system or network infrastructure [10, 11].

Each of these delivery models is above the required hardware and virtualization model as shown in Figure 2.

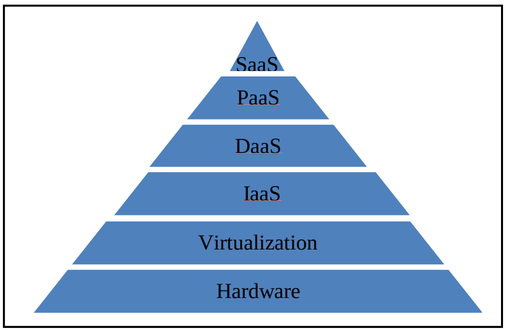

Figure 2. Cloud computing stack layers

No matter what type of delivery model is being used, there are five essential factors or characteristics that have to be satisfied to achieve smooth computing in a cloud computing environment. They are:

a) On-demand self-service: On demand self-service refers to the availability of the required resources (such as CPU power, network etc) as and when the user needs it. Further, this should be without any human intervention [12].

b) Broad network access: As the interacting medium between the different users is the Internet, there should be a broad network access available that allows for the seamless interaction of different applications across different heterogeneous platforms [13].

c) Resource Pooling: A cloud provider should support multi-tenancy of its resources for maximising the efficiency of its infrastructure. For example, it should be able to dynamically assign the required resources to the consumer according to its demand [14].

d) Rapid Elasticity: It should be flexible according to the computing resources required for the customers. For example, there is no up-front commitment and the customers should be able to release the resources once their work is done [15].

e) Measure of Service: There should be a framework that measures the usage of each user according to the resources that are being used by it [16].

To test the performance of hardware or real applications, test and evaluations rules should be defined and implemented to serve as a comparative tool for performance metrics. Instead of having a large investment and lengthy time to use non-reliable providers, benchmarks [17] assist decision makers to save money and choose the cloud providers who fulfil their objectives. Based on the study objectives or research, an appropriate benchmark can be chosen and a targeted application or system is deployed. Then, the results can be analysed using different techniques to obtain the final recommendation. In this paper, we test the stability of performance of different types of Amazon EC2 instances in order to investigate the use of a performance parameter as the main criterion for service level agreements (SLA) between cloud providers and their customers. Before deploying our application on cloud instances, the same application was executed on a local machine and the response time in this experiment was more stable. So, the standard deviation was almost 1.01. But in the cloud environment, the results vary based on the type of EC2 instances. More details about experiment results are discussed in the experiments section.

In our study, we ran a series of experiments on Amazon EC2 cloud over a different number of times. For each time period, we evaluated the response time of five types of Amazon EC2 instances. The main contribution of our study is testing the isolation across the same hardware of virtual machines which are hosted by a cloud provider. There are different ways to evaluate the scalability of cloud providers, for instance, evaluating of throughput of network, disk performance, and capacity of RAM. In this paper, we use the CPU performance as a main parameter for cloud performance, and we measure the execution time of the deployed application over five types of Amazon EC2 instances. We recorded the response time every two hours during several days of experimentation.

The rest of this paper is structured as follows. We discuss related work in Section 2. The methodology of our contribution is presented in Section 3. We present the results and our evaluation in Section 4 and conclude in Section 5.

2. Literature Review

Several studies on the scalability of virtual machines already exist. Most of these studies considered the measurement of performance metrics on the local machines. The background loads of tested machines are controlled to compare the results of performance with a different scale of loads. To the best of our knowledge, to date, no such methodology has been developed to study the performance for cloud providers by considering the use of different metrics of performance. For example, Evangelinos and Hill [18] evaluated the performance of Amazon EC2 to host High Performance Computing (HPC). They use 32-bit architecture for only two types of Amazon instances. In our study, we run various experiments on most types of Amazon EC2 instances. These instances are: small, large, extra large, high CPU, medium, and high CPU extra large instance. Jureta, and Herssens [19] propose a model called QVDP. This model has three functions: specifying the quality level, determining the dependency value, and ranking the quality priority. These functions consider the quality of services from the customers' perspective. However, the performance issues related to cloud resources are not discussed and details are missing regarding the correlation of the quality model with the costing model of services. Cherkasova and Gardner [20] use a performance benchmark to analyse the scalability of disk storage and CPU capacity with Xen Virtual Machine Monitors. They measure the performance parameters of visualization infrastructure that are already deployed in most data centres. But they do not measure the scalability of cloud providers using the visualization resources. However, our proposed work profiles the performance of virtualization resources that are already running on the infrastructure of cloud providers such Amazon EC2 services.

3. Methodology

3.1 Benchmark

We ran Java application on Amazon EC2 over a period of days. We used our benchmark to measure the variations in the performance of CPU for the five types of Amazon EC2 instances. If the collected results show that the execution time of chosen application is stable, then this will provide evidence that a cloud infrastructure is able to run applications which need stability of response time. If the collected results have sizeable variations in response time, then the particular cloud provider is not able to host applications that consider the response time as one of main objectives in the service level agreement (SLA).

3.2 Experiment Setup

We used different types of virtual machines in terms of CPU capacity, RAM size, and bandwidth of disk and network. Table 1 show the features of Amazon EC2 instances that were used in our experiments.

Instance Type	EC unit	Cores	Architecture	Disk (GB)	RAM (GB)
Small	1	1	32	160	1.7
Medium (H-CPU)	5	2	32	350	1.7
Large	4	2	64	850	7.5
Extra Large	8	4	64	1690	15
Extra Large (H-CPU)	20	8	64	1690	7

Table 1. Features of Amazon EC2 instances

There are different uses of cloud computing technology and the results of the performance using different applications are different. The performance comparison is not fair in this case. So, we deploy one Java application on all types of cloud instances and we collect results without changing the scalability of our application. Our goal is to see how the usage changes when the backload is changed in the same machine in the cloud data centre. The proposed metrics to measure the scalability of cloud providers will evaluate throughput of network, disk performance, and capacity of RAM. In this paper, we use the CPU performance as a main parameter for cloud performance; in our future work, we will use the other metrics and evaluate the same types of Amazon EC2 that were used in this paper.

4. Experiment Results

In this section, we compare the response time of selected VMs which are provided by EC2. The performance metric we are measuring does not include the booting and installing time which has various measurements between 80 and 220 seconds. Also, the response time reported does not include the transferring of input and output data. Table 2 shows the 5 samples of performance metrics of EC2 instances.

Small	Medium (H-CPU)	Large	Extra Large	Extra Large (H-CPU)	Local Machine
656	375	110	125	125	360
734	375	125	172	125	360
844	375	109	124	125	359
650	438	172	187	125	360
STDDEV 122.9	STDDEV 23.5	STDDEV 48.9	STDDEV 27.7	STDDEV 7.2	STDDEV 1.1
Average 769.3	Average 383.2	Average 126.9	Average 153.8	Average 129.6	Average 359.8

Table 2. Samples of response time of Amazon EC2 instances

In the performance metrics, the best stability was for the Extra large (H-CPU) type. This is due to the fact that Extra large (H-CPU) has the best resources of CPU power. Figure 3-7 show the stability of the performance on the selected types of VMs.

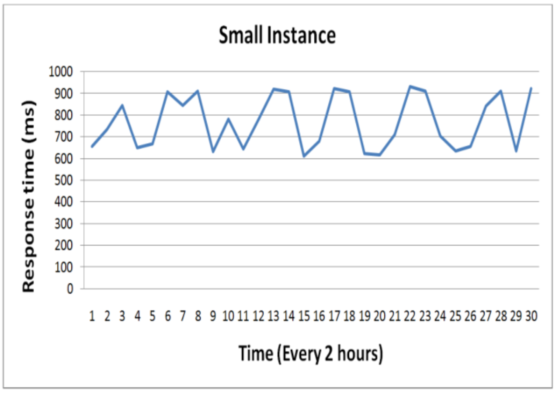

Figure 3. Response time of small instances

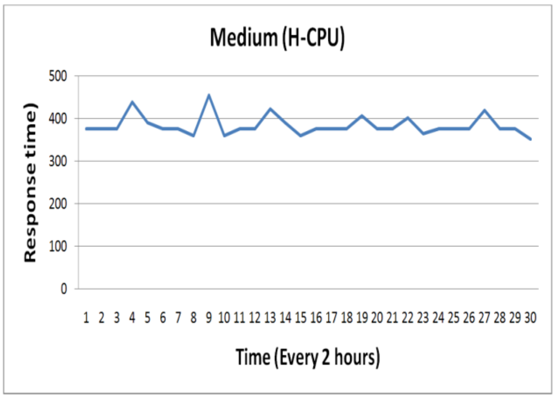

Figure 4. Response time of medium instances

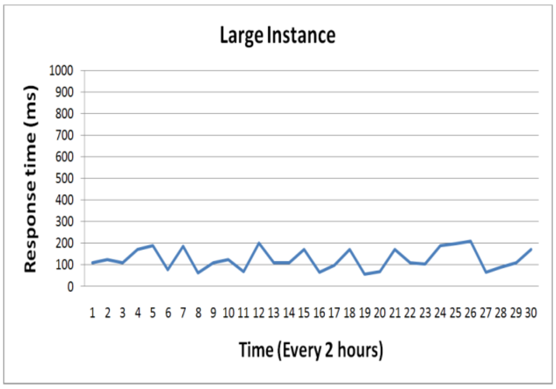

Figure 5. Response time of large instances

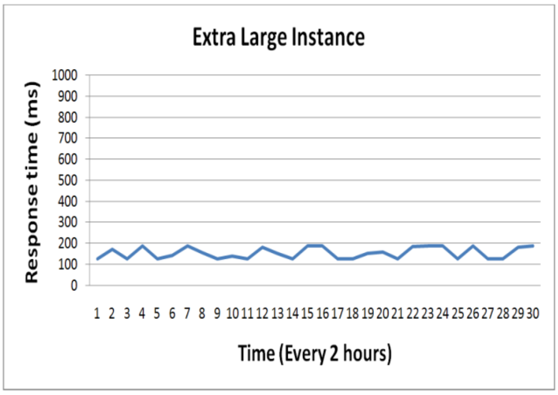

Figure 6. Response time of extra large instances

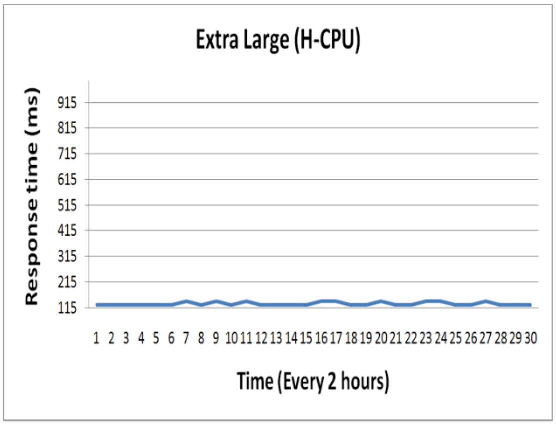

Figure 7. Response time of extra large (H-CPU) instances

5. Conclusion

Cloud computing is a new form of technology, whose infrastructure, developing platform, software, and storage can be delivered as a service in a pay-as-you-use cost model. Intelligent usage of resources in cloud computing may help cloud customers to reduce the large amount of IT investments as well as operational costs. However, for critical business application and more sensitive information, cloud providers must be selected based on high level of performance and trustworthiness. To use cloud services, it is very important to understand the performance of the cloud infrastructure provided by clouds. In this paper, we evaluate the EC2 instances as example to examine the stability of most types of VMs which provided by Amazon. We demonstrate that the performance of Extra large high CPU has the best stability of performance. So, as a service level agreement, response time can be used as a good parameter in the agreement. But for small, large, and extra large instances, it is important to improve the stability of response time before signing any agreement between cloud provider and user.

References

[1] P. Mell and T. Grance, Draft nist working definition of cloud computing, 2009.

[2] J. Napper and P. Bientinesi, Can cloud computing reach the top500?, 2009, pp. 17-20.

[3] Y. Chen, et al., What's New About Cloud Computing Security?, 2010.

[4] R. Buyya, Market-Oriented Cloud Computing: Vision, Hype, and Reality of Delivering Computing as the 5th Utility, 2009, p. 1.

[5] A. Marinos and G. Briscoe, Community cloud computing, CoRR, abs/0907.2485, 2009.

[6] P. T. Jaeger, et al., Cloud computing and information policy: Computing in a policy cloud?, Journal of Information Technology & Politics, vol. 5, pp. 269-283, 2008.

[7] M. Nelson, Building an Open Cloud, Science, vol. 324, p. 1656, 2009.

[8] D. Hilley, Cloud Computing: A Taxonomy of Platform and Infrastructure-level Offerings, 2009.

[9] H. Cai, et al., Customer Centric Cloud Service Model and a Case Study on Commerce as a Service, 2009, pp. 57-64.

[10] D. Cerbelaud, et al., Opening the clouds: qualitative overview of the state-of-the-art open source VM-based cloud management platforms, 2009, pp. 1-8.

[11] J. Muller, et al., Customizing Enterprise Software as a Service Applications: Back-End Extension in a Multi-tenancy Environment, 2009, p. 66.

[12] B. Sotomayor, et al., Virtual infrastructure management in private and hybrid clouds, IEEE Internet Computing, vol. 13, pp. 14-22, 2009.

[13] D. Nurmi, et al., The eucalyptus open-source cloud-computing system, 2009, pp. 124-131.

[14] M. Zeller, et al., Open standards and cloud computing: Kdd-2009 panel report, 2009, pp. 11-18.

[15] T. Dillon, et al., Cloud Computing: Issues and Challenges, 2010, pp. 27-33.

[16] J. Nunamaker Jr, et al., Systems development in information systems research, Journal of Management Information Systems, pp. 89-106, 1990.

[17] P. Donohoe, A Survey of Real-Time Performance Benchmarks for the Ada Programming Language, ed: Citeseer, 1987.

[18] C. Evangelinos and C. Hill, Cloud Computing for parallel Scientific HPC Applications: Feasibility of running Coupled Atmosphere-Ocean Climate Models on Amazon's EC2, ratio, vol. 2, p. 2.34, 2008.

[19] I. Jureta, et al., A comprehensive quality model for service-oriented systems, Software Quality Journal, vol. 17, pp. 65-98, 2009.

[20] L. Cherkasova and R. Gardner, Measuring CPU overhead for I/O processing in the Xen virtual machine monitor, 2005, p. 24.