Evaluating Eucalyptus Virtual Machine Instance Types: A Study Considering Distinct Workload Demand

Erica Teixeira Gomes de Sousa, Paulo Romero Martins Maciel, Erico Moutinho Medeiros, Débora Stefani Lima de Souza, Fernando Antonio Aires Lins, Eduardo Antonio Guimaraes Tavares
Center of Informatics
Federal University of Pernambuco
Recife, Brazil
{etgs,prmm,emm2,dsls,faal2,eagt}@cin.ufpe.br

***Abstract*—Cloud computing paradigm provides virtualized computing resources as a service on demand. This paradigm allows companies focus on their business issues rather than conceiving and managing complex infrastructures. As a consequence, performance evaluation of cloud computing infrastructures has been receiving considerable attention by service providers as a prominent activity for improving service quality infrastructure planning, and selection of software platforms (e.g., Eucalyptus). This paper presents the performance evaluation of virtual machines on Eucalyptus platform considering different workloads. This study provides insights about Eucalyptus system infrastructure suitability for applications with high processing and storage performance requirements. This work provided the evaluation of virtual machine instances types from a private cloud, which was configured with Eucalyptus platform.**

Keywords-cloud computing; eucalyptus platform; performance evaluation.

I. INTRODUCTION

Cloud computing is a combination of technologies that have been developed over the last several decades, which includes virtualization, grid computing, cluster computing and utility computing. Due to market pressure, cloud computing technologies have rapidly evolved in order to let users focus on business aspects rather than complex infrastructures issues, hence fostering business competitiveness [1][2].

Currently, cloud providers provide guarantees on their service levels and when service failures occur, they only offer to refund their customers regarding the infrastructure outages. However, service providers are not inclined to pay penalties that would refund customers for loss of business revenue [3][4]. Cloud providers are not only required to supply correct services but, also, to meet their expectations in the context of performance. Indeed, cloud computing services have been massively expanding, thus demanding companies to offer reliable services, high availability, scalability and security at affordable costs. In such a case, performance evaluation is a prominent activity for improving service quality, infrastructure planning, and for tuning system components [5][6].

Some software systems, such as credit and debit processing applications require different performance levels, quality of services, reliability, and security, which are generally not guaranteed by a public cloud. In these clouds, computing resources are shared with other companies. These companies do not have any knowledge or control of where the applications run. Private cloud is an alternative to companies that need more control over that data. In private clouds, data-center resources of a company are controlled by company's IT staff [7][8]. Eucalyptus is an open source cloud computing platform that implements infrastructure as a service (IaaS) on a collection of server clusters, and such platform allows the creation of private clouds [9].

Different works [10][11] propose an evaluation of the performance of various public cloud computing infrastructures for suitability in scientific applications. Some other papers [12][13] present the performance evaluation of public and private cloud computing storage resources.

In this work, we evaluate the performance of different virtual machines types on the Eucalyptus platform [9] to determine its suitability for applications that demands different processing and storage performance. More specifically, the aim of this work is to evaluate these virtual machines according to a specified workload. Different from presented papers, this work proposes the evaluation of the Eucalyptus virtual machine instance types considering distinct workload demand.

This paper is structured as follows: Section 2 presents related works on performance evaluation in cloud computing environments. Section 3 introduces basic concepts on Eucalyptus platform and Performance Evaluation. Section 4 presents the adopted methodology for performance evaluation and Section 5 shows a real case study. Finally, Section 6 presents concluding remarks and presents future works.

II. RELATED WORKS

In the last few years, some works have been conducted to evaluate performance of public cloud computing infrastructures for scientific applications. Ostermann et al. [10] present a performance analysis of Amazon EC2 platform for scientific computing using benchmarks. Similarly, in [11], the authors propose a performance evaluation of four commercial cloud computing services

 ISBN: 978-1-61208-216-5

using benchmarks. These clouds computing are Amazon EC2, GoGrid (GG), Elastic Hosts (EH) and Mosso [11]. These papers compare only the performance and cost of clouds with scientific computing alternatives such as grids and parallel production infrastructures. The result is that the current cloud computing services are insufficient for scientific computing at large, but it may still be a good solution for the scientists who need resources instantly and temporarily [10][11].

Some papers focus on performance evaluation of storage resources from public and private clouds for scientific application. In [12], the Eucalyptus Platform is tested in a variety of configurations to determine its suitability for applications with high I/O performance requirements, such as the Hadoop MapReduce framework for data-intensive computing. Applications running in the Eucalyptus cloud computing framework suffer from I/O virtualization bottlenecks in KVM and Xen. The default configurations that use fully virtualized drivers and on-disk files perform poorly out of the box. Even using the best default configuration (with the Xen hypervisor), the guest domain can achieve only 51% and 77% of non-virtualized performance for storage writes and reads, respectively [12]. In [13], a public cloud platform and a private cloud platform are evaluated. The authors select the Amazon as the public cloud platform and the Magellan cloud testbed as the private cloud computing. Such a work compares the I/O performance using IOR benchmarks. The I/O performance results clearly highlight that I/O can be one of the causes for bottleneck on virtualized cloud environments. Performance in VMs is lower than on physical machines, which may be attributed to an additional level of abstraction between the VM and the hardware [13].

Differently from previous works, this paper presents the performance evaluation of different virtual machines on Eucalyptus platform, focusing on processing and storage infrastructures.

III. PRELIMINARIES

This section presents a summary of concepts for a better understanding of this work. Initially, an overview of Eucalyptus platform is provided. Finally, performance evaluation concepts are presented.

A. *Eucalyptus Platform*

Eucalytpus is an open-source cloud computing platform that allows the creation of private clusters in enterprise datacenters [14]. Eucalyptus provides API compatibility with the most popular commercial cloud computing infrastructure, namely, Amazon Web Services (AWS), which allows management tools to be adopted in both environments. This framework is designed for compatibility across a broad spectrum of Linux distributions (e.g., Ubuntu, RHEL, OpenSUSE) and virtualization hypervisors (e.g., KVM, Xen). Figure 1 shows the Eucalyptus architecture.

Eucalyptus system is composed of several components that interact through interfaces [14]. There are five components, each with of which its own Web-service interface, that comprise a Eucalyptus system [15]:

- **Cloud Controller (CLC).** The CLC is the entry-point into the cloud for users and administrators. It queries node managers for information about resources, performs high-level scheduling decisions, and implements them by making requests to cluster controllers.
- **Cluster Controller (CC)**. The CC acts as a gateway between the CLC and individual nodes in the data center. This component collects information on schedules and execution of virtual machine (VM) on specific node controllers, and manages the virtual instance network. The CC must be in the same Ethernet broadcast domain as the nodes it manages.
- **Node Controller (NC)**. The NC contains a pool of physical computers that provide generic computation resources to the cluster. Each of these machines contains a node controller service that is responsible for controls the execution, inspection, and terminating of virtual machine (VM) instances. This component also configures the hypervisor and host OS as directed by the CC. The node controller executes in the host domain (in KVM) or driver domain (in Xen) [9][15].
- **Storage Controller (SC)**. The SC is a put/get storage service that implements Amazon's S3 interface, providing a mechanism for storing and accessing virtual machine images and user data.

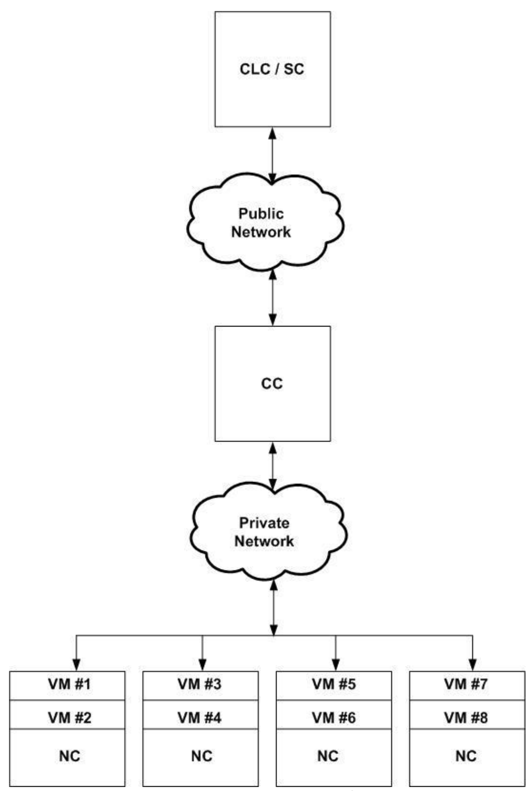

Figure 1. Eucalytus platform.

Eucalyptus platform supports different virtual machine (VM) [9][15]. The supported virtual machines and the respective characteristics are presented in Table 1. The computational resources can be allocated to these virtual machines according to the client demand.

TABLE I. VIRTUAL MACHINE INSTANCE TYPES

VM	CPU (Core)	Memory (MB)	Disk (GB)
m1.small	1	192	2
c1.medium	1	256	5
m1.large	2	512	10
m1.xlarge	2	1024	20
c1.xlarge	4	2048	20

B. *Performance Evaluation*

Performance evaluation consists of a technique set classified as those based on measurement and based on modeling, which provide means for deciding about suitable configurations concerning further customers' demands, fluctuations and attaining assured service levels [16].

Modeling is usually adopted in early stages of the design process, when actual systems are not available for measurement. Measurement is utilized for understanding systems that are already built or prototyped. Measurement is an essential activity for tuning the systems, validating the performance models, and for improving the design of future systems [5][17].

A variety of different benchmark programs have been developed over the years to measure the performance of many different types of computer systems in different application domains. A natural benchmark consists of programs that mimic a real workload. Synthetic benchmark programs are artificial programs. These programs perform a mix of operations that are carefully chosen to match the relative mix of operations observed in some class of application [6][16].

Bonnie++ and LINPACK [5][16][18] are benchmarks adopted in this paper to provide workload to disk and processor components, respectively. Bonnie++ is a benchmark suite that executes a number of hard drive and file system performance tests [18]. LINPACK is a benchmark that solves dense system of linear equations in double precision and is commonly used for performance evaluation of parallel computers as a cluster. LINPACK is a benchmark that allows defining the size of the system of linear equations in order to evaluate the performance computer systems [5].

Linux monitoring tools IOstat and MPstat [19] are adopted in this work to collect professor and disk figures, respectively.

IV. PERFORMANCE EVALUATION METHODOLOGY

This section presents the methodology used for performance evaluation adopted for analyzing the Eucalyptus platform. Figure 2 shows the activity diagram of the adopted methodology.

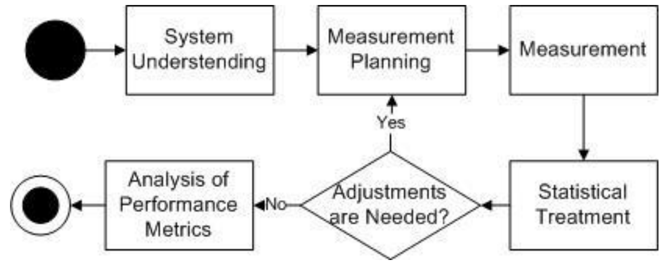

Figure 2. Methodology.

The methodology consists of five activities, which are system understanding, measurement planning, measurement, analysis of performance metrics and statistical analysis. The first activity concerns understanding the system, its components, their interfaces and interactions. This activity should provide the set of metrics that should be evaluated. Among such metrics, some might be highlighted such as utilization, average service time, average response time, average queue time, average queue length.

The second activity results in a document that describes how the measurement should be performed, the tools calibration, the frequency of data collection and how to store the measured data.

The measurement activity (see Figure 3) consists of five steps, which were implemented through a script. These steps are described below.

The first step instantiates the virtual machines. The second step starts the performance monitoring tools IOstat and MPstat on virtual machines. The third step configures the benchmarks Bonnie++ and LINPACK on virtual machines. The fourth step executes the benchmarks. These benchmarks are the workload adopted. When benchmarks execution end, in the fifth step, the monitoring processing finishes. Then, the virtual machines are shutdown. Next, logs with the measured data are created. After, the measurement process is restarted.

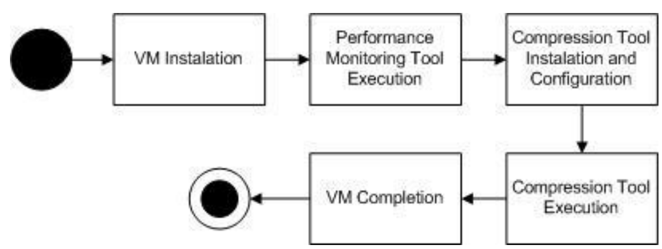

Figure 3. Measurement activity.

In each measurement process, the monitoring tools are started before the benchmarks execution. These measured data must be removed; therefore, the fourth activity analyzes the measured data.

Finally, the fifth activity applies statistical methods in measured data with the aim of providing accurate information about the evaluated system.

V. EXPERIMENTAL RESULTS

This section presents the conducted experiments to evaluate bottlenecks regarding processing and storage

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-216-5

resources running over Eucalyptus system virtual machines. This case study allows the performance evaluation of default virtual machine instances types from a private cloud configured with the Eucalyptus system. The performance of the processing and storage infrastructures is evaluated with benchmarks.

These experiments aim to evaluate the performance of cloud infrastructure presented in Figure 1 assuming a workload based on benchmarks Bonnie++ and LINPACK. These experiments consisted of some steps, which are described in Section IV, as shown in Figure 2.

In this case study, Cloud Controller (CLC), Cluster Controller (CC) and Storage Controller (SC) are running on the same computer, whereas Node Controllers (NC) are distributed on different computers. The front-end node and the back-end nodes were equipped with Intel^{R} Core^{TM} Duo CPU E6550, 2.33GHz, 2GB DDR2 RAM, 160 GB Hard Disk and 100MB Ethernet interface. These computers were configured to run the CLC, CC, SC and NC services. The front-end node was configured to run the CLC, CC and SC services. The back-end nodes were configured to run the NC service and all virtual machine images.

In this case study, the virtual machine environment is based on Eucalyptus system 2.0.0 and Ubuntu Enterprise Cloud 10.10. The Eucalyptus was configured with kernel-based virtual machines (KVM) as a hypervisor.

The virtual machine types evaluated were m1.small, c1.medium, m1.large and m1.xlarge (see TABLE I). These virtual machines have different processing and storage infrastructures, which can be allocated and released according to users demand.

The scenarios evaluated in the performance experiments are described in TABLE II. This table shows the types and number of virtual machines in each scenario analyzed. The numbers of virtual machines for each scenario varies according to the processing and storage resources of these machines and node controllers processing and storage infrastructures.

TABLE II. SCENARIOS

Scenario	VM Number	VM Types
1	8	m1.small
2	8	c1.medium
3	4	m1.large
4	4	m1.xlarge

As previously mentioned, we adopt the benchmarks LINPACK and Bonnie++ and the performance monitoring tools IOstat and MPstat in order to obtain the desired performance metrics. TABLE III shows the relationship between the benchmarks adopted to provide workload to processor and disk and monitoring tools adopted in order to provide processor and disk metrics.

TABLE III. BENCHMARK AND MONITORING TOOL

Resource	Benchmark	Tool
Disk	Bonnie++	IOstat
Processor	LINPACK	MPstat

After setting up and stabilizing the environment, measurements of performance metrics were initiated through the IOstat and MPstat. During the measurements, processes that are not strictly necessary for the experiments were removed so as to avoid interference in the collected data [5].

Measurements of processor metrics were performed for a period of 12 hours with an interval of 1 minute between data collections. The processor experiments considered the LINPACK benchmark as workload, where the number of linear equations adopted to evaluate the system was N = 1000 [5].

Measurements of disk metrics were performed for a period of 24 hours considering reading and writing tests in which files with 512MB and 1GB were considered. On the other hand, reading and writing tests considering files with 1.5GB and 2GB sizes were performed for a period of 12 hours. Each sample was collected considering an interval of 1 minute between data collections. The Bonnie++ benchmark stresses the system by performing reading and writing operations on a file system. For this experiment, files with 512MB, 1GB, 1.5GB and 2GB sizes were created for all evaluated scenarios. These files are created when the benchmark Bonnie++ is running. Furthermore, these files aim to evaluate the performance of the disk in the scenarios described in TABLE II.

These collected data were stored in logs generated through the scripts. The collected data were stored on a disk partition isolated from the measuring environment in order to prevent the measured data from being affected.

These collected data were statistically analyzed to remove possible outliers [20]. TABLE IV presents the execution times of the performance tests. The processor performance tests were performed during 12 hours and the disk performance tests occurred according to files sizes adopted in reading and writing testes.

TABLE IV. PERFORMANCE TEST

Benchmark	File Size	Execution Time (hour)
Bonnie++	512 MB	24
Bonnie++	1 GB	24
Bonnie++	1.5 GB	12
Bonnie++	2 GB	12
LINPACK	-	12

Table V shows the performance metrics evaluated in this case study.

TABLE V. PERFORMANCE METRICS

Resource	Metric
Disk	Utilization, Service Time, Response Time
Processor	Utilization

 ISBN: 978-1-61208-216-5

Figure 4 presents the processor utilization for each virtual machine (see TABLE II). This work adopts 80% as threshold [21].

Processor results reveal that the processor utilization of virtual machines exceed 80% for all scenarios [21], hence these processing infrastructures should be updated or the processor resource of the node controllers should be replaced as a preventive measure for taking into account further workload demands.

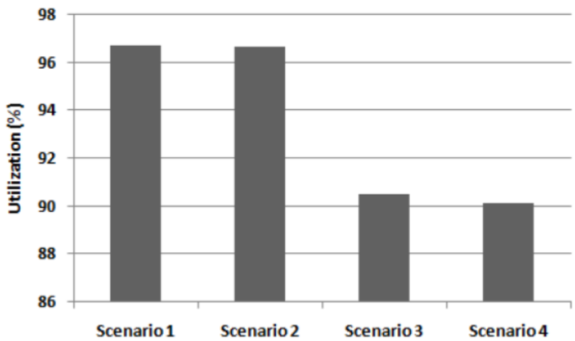

Figure 4. Processor.

Figure 5, Figure 6 and Figure 7 show the results of disk metrics which are utilization, average service time and average response time, respectively. The values of these disk metrics vary proportionally to the testing of reading and writing considering files with 512MB and 1GB sizes. These tests were performed for all scenarios evaluated (see TABLE II).

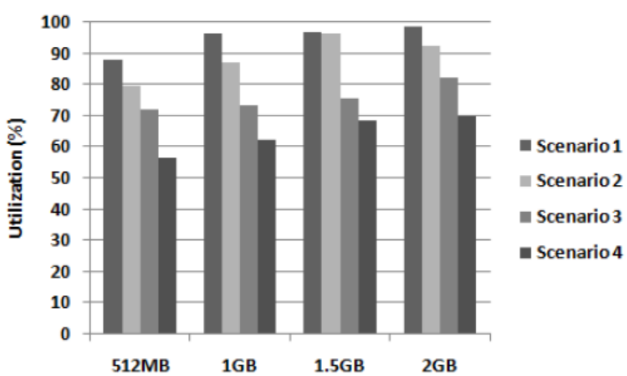

Figure 5. Utilization.

Figure 5 shows a percentage reduction in the disk utilization level when analyzing the Scenarios 1 to 4 for all reading and writing tests (files with 512MB, 1GB, 1.5GB and 2GB sizes). This metric indicates the percentage of time that the disk was used. If the disk utilization is high, the resource must be carefully evaluated. The threshold for this disk metric is close to 100%. Hence, this metric should be investigated if it is close to 100% [21].

These disk results reveal that the disk utilization level of Scenario 1 exceed 90% when considering tests of reading and writing adopting files with 1GB, 1.5GB and 2GB sizes. In a similar way, the disk utilization level of Scenario 2 exceed 90% when considering tests of reading and writing adopting files with 1.5GB and 2GB sizes. These results demonstrate that the storage infrastructure must be carefully analyzed to avoid a bottleneck and hence degradation in service levels [21].

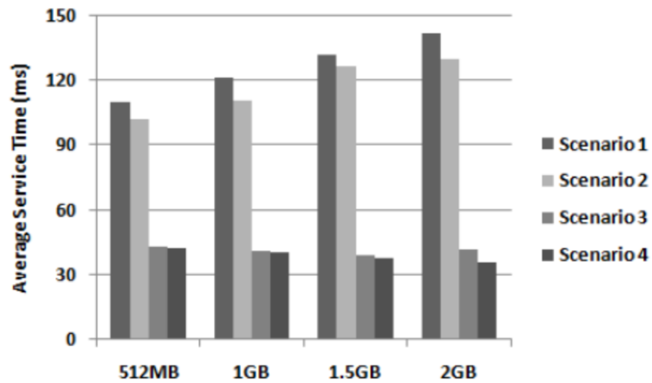

Figure 6. Average service time.

Figure 6 presents a decrease in the average service time (in milliseconds) for Scenarios 1 to 4 and all reading and writing tests performed by benchmark Bonnie++. This metric describes how long time the disk is taking to fulfill the requests. When more time is spent on fulfilling the requests, slower is the disk controller. It is recommended that the values of these metric be less than 270 ms [21]. The results show that the storage resources do not exceed the threshold of this disk metric.

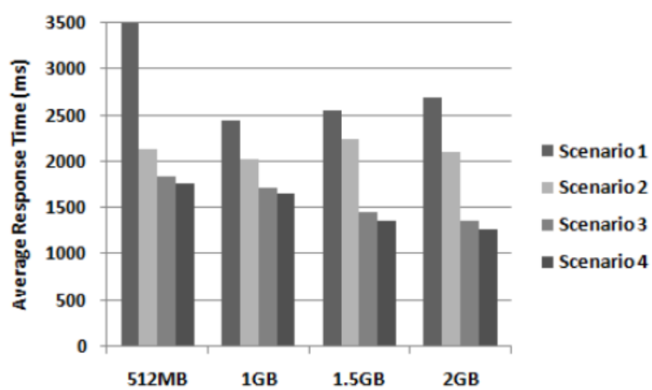

Figure 7. Average response time.

Figure 7 presents a decrease in the average response time (in milliseconds) for Scenarios 1 to 4 when considering all reading and writing tests (files with 512MB, 1GB, 1.5GB and 2GB sizes). This disk metric includes the time spent by the requests in queue and the time spent servicing them. If the average response time is high, the resource must be carefully evaluated. The threshold for this disk metric is 2.7 seconds [21]. The average response time of Scenario 1 exceed 2.7 seconds when considering tests of reading and writing performed by benchmark Bonnie++ adopting files with 512MB and 2GB sizes [21].

Processing infrastructures results had lower performance, considering Scenarios 1 and 2, in relation to the others, for the same workload. Scenario 1 shown a storage with lower performance in comparison to the others scenarios, for the same workload.

This analysis permits the planning of processing and storage infrastructure of the private cloud in the enterprise.

VI. CONCLUSIONS AND FUTURE WORK

This work presented the performance evaluation of different virtual machines on Eucalyptus platform taking into account workload based on benchmarks.

This paper analyzed the performance of critical levels of virtual machine instances types on private cloud, which is configured with Eucalyptus system. This analysis permits sustain the quality of service and prevents the performance degradation related to the workload fluctuations in private clouds. In addition, this performance analysis is intended to support suitable hardware and software configurations for ensuring performance agreements of applications with high level of processing and storage requirements.

The results allowed evaluation of performance figures, such as disk response time, disk service time, disk utilization and processor utilization, for planning the Eucalyptus system processing and storage infrastructures.

Other performance issues related to Eucalyptus system can be studied and analyzed as well as other metrics than those discussed in the paper. As future work, we intend to analyze the performance of processing and storage infrastructures of virtual machine instances types from a private cloud, considering credit and debit transactions (Electronic Funds Transfer) as workload.

REFERENCES

[1] Velte, A., Velte, T., Elsenpeter, R., and Babcock, C., Cloud computing: a practical approach. McGraw-Hill, 2010.

[2] Chee, B. and Franklin Jr, C., Cloud computing: technologies and strategies of the ubiquitous data center. CRC Press, 2009.

[3] Hugos, M. H. and Hulitzky, D., Business in the Cloud: What Every Business Needs to Know About Cloud Computing. Wiley, 2010.

[4] Chorafas, D. and Francis, T., Cloud computing strategies. CRC Press, 2011.

[5] Jain, R., The art of computer systems performance analysis, vol. 182. John Wiley & Sons New York, 1991.

[6] Menasce, D. A., Almeida, V. A. F., Dowdy, L. W., and Dowdy, L., Performance by design: computer capacity planning by example. Prentice Hall, 2004.

[7] Shroff, G., Enterprise Cloud Computing: Technology, Architecture, Applications. Cambridge Univ Pr, 2010.

[8] Rosenberg, J. and Mateos, A., The cloud at your service. Manning, 2010.

[9] Johnson, D., and Murari, K., Raju, M., Suseendran, R. B., and Girikumar, Y., "Eucalyptus Beginners Guide - UEC Edition," 2010.

[10] Ostermann, S., Iosup, A., Yigitbasi, N., Prodan R., Fahringer, T., and Epema, D., "A Performance Analysis of EC2 Cloud Computing Services for Scientific Computing," pp. 115–131, 2010.

[11] Iosup, A., Ostermann, S., Yigitbasi, N., Prodan, R., Fahringer, T., and Epema, D., "Performance Analysis of Cloud Computing Services for Many-Tasks Scientific Computing," IEEE Transactions on Parallel and Distributed Systems, pp. 1–16, 2010.

[12] Shafer, J., "I/O virtualization bottlenecks in cloud computing today," in Proceedings of the 2nd conference on I/O virtualization, pp. 5–5, USENIX ssociation, 2010.

[13] Ghoshal, D., Canon, R., and Ramakrishnan, L., "Understanding I/O performance of virtualized cloud environments", The Second International Workshop on Data Intensive Computing in the Clouds (DataCloud-SC11), 2011.

[14] Amazon Web Services, "Eucalyptus open-source cloud computing infrastructure - an overview. technical report, eucalyptus, inc.," 2011.

[15] Nurmi, D., Wolski, R., Grzegorczyk, C., Obertelli, G., Soman, S., Youseff, L., and Zagorodnov, D., "The eucalyptus open-source cloud-computing system," in Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, pp. 124–131, IEEE Computer Society, 2009.

[16] Lilja, D. J., Measuring computer performance: a practitioner's guide. Cambridge Univ Pr, 2005.

[17] John, L. K. and Eeckhout, L., Performance evaluation and benchmarking. CRC Press, 2006.

[18] Coker, R., "The bonnie++ benchmark," 2001. URL http://www. coker. com. au/bonnie+ Last visit in April 2, 2012.

[19] Godard, S., Sysstat:System performance tools for the Linux OS. 2004.

[20] Montgomery, D. and Runger, G., Applied statistics and probability for engineers. Wiley, 2010.

[21] Ciliendo, E. and Kunimasa, T., Linux Performance and Tuning Guidelines. ibm.com/redbooks, 2007.