CloudMeasure: A REPOSITORY OF PERFORMANCE MEASURES FOR PERFORMANCE ANALYSIS OF CLOUD COMPUTING SYSTEMS

Research Proposal

Luis Eduardo Bautista Villalpando[1], Alain April[2]

[1]Department of Electronic Systems, Autonomous University of Aguascalientes, Aguascalientes, Mexico; [2]Department of Software Engineering and Information Technology, ETS – University of Quebec, Montreal, Canada.
Email: lebautis@correo.uaa.mx, alain.april@etsmtl.ca

***Abstract*— One of the main issues in Cloud Computing is the lack of information which helps to understand and define concepts of assurances of availability, reliability and liability in Cloud Computing Systems (CCS). Concepts such as price, performance, time to completion (availability), probability of failure and liability are key to being able to produce a comparison service, in order to establish Service Level Agreements (SLA) or design better mechanisms to improve the performance in CCS. This work proposes the design of a repository of performance attributes, which provides information and tools to facilitate the design, validation, and comparison of performance analysis models and algorithms for CCS. The purpose of this repository is to help to establish attribute–performance relationships relating to specific applications with relatively well-known demands on systems to be able to determine how comparison services may be formulated. The design of the CloudMeasure repository is based on the Performance Measurement Framework for Cloud Computing which defines the basis for the measurement of Cloud Computing concepts that are directly related to performance and have been identified from international standards such as ISO 25010.**

***Index Terms*— Cloud Computing, Measurement, Performance, Repository, ISO 25010 Quality Model, Maintenance**

I. INTRODUCTION

Cloud Computing (CC) is an emerging technology aimed at processing and storing very large amounts of data over the Internet. According to the ISO subcommittee 38 – the study group on Cloud Computing [1], CC is a paradigm for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable cloud resources accessed through services, that can be rapidly provisioned and released with minimal management effort or service provider interaction.

This normalization study group [2] mentions that cloud services are categorized in three service models: 1) Infrastructure as a Service (IaaS), 2: Platform as a Service (PaaS), and 3: Software as a Service (SaaS). These three service models include all the technical resources that clouds need in order to process information; like software, hardware, and network elements. For example, the service model that relates the most to the software engineering community is the SaaS

model while the IaaS model is most related to hardware architectures and virtualization. Software engineers focus on software components, and customers use an IT provider's applications running on a cloud infrastructure to process information according to their processing and storage requirements. One of the main characteristics of IaaS model is that customers do not manage or control the underlying Cloud infrastructure (including network, servers, operating systems, and storage), except for limited user-specific application configuration settings.

II. PROBLEM TO BE SOLVED

One of the most important challenges in delivering Cloud Services is to ensure that they are fault tolerant and minimize anomalies which can degrade its services or impact their quality, and even their availability. According to Coulouris [3], a failure occurs in a distributed system (DS), like a CC system (CCS), when a process or a communication channel departs from what is considered to be its normal or desired behaviour. An anomaly is different, in that it slows down a part of a CCS without making it fail completely, impacting the performance of tasks within nodes, and, consequently, of the system itself.

Performance analysis models (PAM) for CCS should propose a means to identify and quantify "normal cluster behaviour," which can serve as a baseline for detecting possible failures and anomalies in the computers (i.e. nodes in a cluster) that may impact the overall cloud performance. To achieve this goal, methods are needed to collect the necessary base measures specific to CCS performance, and analysis models must be designed to determine the relationships that exist among these measures. The ISO International Vocabulary of Metrology (VIM) [4] defines a measurement method as a generic description of a logical organization of operations used in measurement, and an analysis model as an algorithm or calculation combining one or more measures obtained from a measurement method to produce evaluations or estimates relevant to the information needed for decision making.

III. PROPOSED SOLUTION

An important aspect in the creation of the above models is its data requirements. Data is necessary to carry on experiments, simulating different scenarios, in order to select the best models that fit our requirement. Because of this, it is necessary to have access to performance data repositories which help in the process of creation of different models that subsequently can be implemented in live CCS. Unfortunately, currently there are no CC performance repositories that help in the design of such models. We have found some initiatives that make available data from CCS failure (for example: the Google Cluster Data [5] and the Failure Trace Archive (FTA) [6]). A failure trace is a data set collected from different distributed systems which contains records of their availability represented in time series.

This proposal presents the CloudMeasure project which aims to develop and make available a distributed public repository of performance attributes for the performance analysis of CCS. The purpose of this research project is to provide data and tools to facilitate the design, validation, and comparison of performance models and algorithms for CCS. One important aspect of this project is to define the group of CCS elements and the attributes that will constitute the repository. For instance, one group of elements could be the Job history in CCS and some of their attributes could be the number of success tasks executed or the time taken to process them.

IV. THE CLOUDMEASURE REPOSITORY

The repository of performance measures provided by the CloudMesure research project can be used for the performance analysis of CCS. The purpose of this project is to provide information and tools to facilitate the design, validation, and comparison of performance models and algorithms for CCS. One of the main reasons for the creation of the CloudMeasure project is the current lack of information that could help in understanding and defining how to measure availability, reliability and liability of CCS. Actual measurement of concepts such as price, performance, time to completion (availability), likelihood of completion (probability of failure) and penalty (liability) are key to being able to compare services or to establish Service Level Agreements (SLA) for CCS.

According to Li [7], commercial CCS currently enable the capture of price–performance information relating to specific applications with relatively well-known demand patterns. This allows the user to gain useful information to compare the service between suppliers. Comparisons can become complex as they can depend on both the performance requirements of the user, the current availability of the system, as well as the price the user can afford. According to Gangadharan [8], the pricing of Cloud Computing services is currently associated with differentiated levels of service based on varying capacity

of memory, computing units used, and types of platforms. The pricing also varies with respect to the choice of operating systems and the geographical location of the user. The criteria for pricing of Cloud Services can also be based on hourly usage, CPU cycle usage, or other usage approach. In addition Gangadharan mentions that pricing of infrastructure Cloud Services depends upon levels of use, layers of service, or a mix of these. Thus, the CloudMeasure project could provide a useful tool for maintainers, users and developers to help to create performance models to gain knowledge that can contributes to understand an SLA as well as help them measure the performance of CCS. In particular, the CloudMeasure repository will contain the following:

- Performance attributes of CCS, differing in scale and granularity which contributes to create models to analyse concepts as availability, liability, capacity, etc.
- A standard format for performance measures, scripts and tools for analyzing performance of CCS

V. CONCLUSIONS

The CloudMeasure repository is a research project to contribute to the sharing of CCS performance analysis. This project proposes a sound and flexible approach which can be useful to support the analysis of the quality characteristics of CCS during the development, maintenance and operational stages.

Using the information contained in this repository will allow end users to better understand and compare the relationship between performance attributes of different CCS. Moreover, this repository will help to design, validation and compare performance models (and algorithms) which can be used during the design of service licence agreements (SLA). One of the main issues in formulating SLA is how to capture price–performance information relating to specific applications with relatively well-known demands on systems to be able to determine how such a comparison service may be formulated. In this way, the CloudMeasure repository would be an efficient tool, which helps in the creation of models to determine the price-performance comparisons.

Further work is needed to review and increase the data within the CloudMeasure repository as well as tools for creating performance analysis models, which contribute to analyze the performance of real CCS that could contribute to validating proposed methodologies.

REFERENCES

1. ISO/IEC JTC 1 SC38, “Cloud Computing Overview and Vocabulary” International Organization for Standardization, Geneva, Switzerland, October 2012.
2. ISO/IEC JTC 1 SC38, “Study Group Report on Cloud Computing,” International Organization for Standardization, Geneva, Switzerland, November 2011.
3. Coulouris, G., Dollimore, J., and Kindberg, T., "Distributed Systems Concepts and Design," Addison-Wesley, 4th edition, ISBN 0-321-26354-5, Pearson Education Edinburgh, 2005.
4. ISO/IEC Guide 99-12, “International Vocabulary of Metrology – Basic and General Concepts and Associated Terms, VIM,” International Organization for Standardization ISO/IEC, Geneva, Switzerland, 2007.
5. Google Inc., “Google Cluster Data.” [Online]. Available:http://code.google.com/p/googleclusterdata/
6. D. Kondo, B. Javadi, A. Iosup, and D. Epema., “The Failure Trace Archive: Enabling comparative analysis of failures in diverse distributed systems.” In CCGRID, pages 1–10, 2010.
7. Li, B., Gillam, L., and O’Loughlin, “Towards Application-Specific Service Level Agreements: Experiments in Clouds and Grids,” Cloud Computing: Principles, Systems and Applications, Computer Communications and Networks, Springer-Verlag, 2010, doi: 10.1007/978-1-84996-241-4_21.
8. Gangadharan, G. and Parrilli, D.M., “Service Level Agreements in Cloud Computing: Perspectives of Private Consumers and Small-to-Medium Enterprises”, Cloud Computing for Enterprise Architectures, Computer Communications and Networks, Springer-Verlag London Limited 2011, doi: 10.1007/978-1-4471-2236-4_11.