PERFORMANCE EVALUATION FOR VIRTUAL SERVER MANAGEMENT IN CLOUD COMPUTING

R.Sudha Abirami[1], M.Sultan[2], T.Ameela[3], N.Muruganandham[4]

[1]*Dept of CSE, St. Michael College of Engg & Tech/Anna University, Chennai. (India)*
[2]*Dept of CSE, St. Michael College of Engg & Tech/Anna University, Chennai. (India)*
[3]*Dept of CSE, St. Michael College of Engg & Tech/Anna University, Chennai. (India)*
[4]*Dept of CSE, St. Michael College of Engg & Tech/Anna University, Chennai. (India)*

ABSTRACT

Cloud Computing is defined as a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers. It is based on service-level agreements that established between service providers and consumers. Cloud Computing opens up many new possibilities for Internet application developers. The VMware is used to integrate the virtual servers which are used for performance analysis.

Keywords: Cloud Computing, VMware, virtualization, CloudSim, SaaS

I INTRODUCTION

The Cloud computing is Internet-based computing, whereby shared servers provide resources, software, and data to computers and other devices on demand, as with the electricity grid. Cloud computing is a natural evolution of the widespread adoption of virtualization, service-oriented architecture and utility computing. Details are abstracted from consumers, who no longer have need for expertise in, or control over, the technology infrastructure "in the cloud" that supports them.

Server was underutilized because peak traffic happens only at specific times. With the advent of the Cloud, deployment and hosting became cheaper and easier with the use of pay-peruse flexible elastic infrastructure services offered by Cloud providers. Because several Cloud providers are available, each one offering different pricing models and located in different geographic regions, a new concern of application developers is selecting providers and data center locations for applications.

Cloud Analyst helps developers with insights in how to distribute applications among Cloud infrastructures and value added services such as optimization of applications performance and providers incoming with the use of Service Brokers.

II. LITERATURE SURVEY

2.1. Cloud Analyst: A Cloud sim-based Visual Modeller for Analysing Cloud Computing Environments and Applications.

IJARSE

Advances in cloud computing opens up many new possibilities for internet applications developers. Previously, a main concern of Internet applications developers was deployment and hosting of applications, because it [9] required acquisition of a server with a fixed capacity able to handle the expected application peak demand and the installation and maintenance of the whole software infrastructure of the platform supporting the application. Furthermore, server was underutilized because peak traffic happens only at specific times.

With the advent of the Cloud, deployment and hosting became cheaper and easier with the use of pay-peruse flexible elastic infrastructure services offered by Cloud providers. Because several Cloud providers are available, each one offering different pricing models and located in different geographic regions, a new concern of application developers is selecting providers and data center locations for applications. However, there is a lack of tools that enable developers to evaluate requirements of large-scale Cloud applications in terms of geographic distribution of both computing servers and user workloads. To fill this gap in tools for evaluation and modelling of Cloud environments and applications, we propose Cloud Analyst. It [1] was developed to simulate large-scale Cloud applications with the purpose of studying the behaviour of such applications under various deployment configurations. Cloud Analyst helps developers with insights in how to distribute applications among Cloud infrastructures and value added services such as optimization of applications performance and providers incoming with the use of Service Brokers.

2.2 Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities

This keynote paper: presents a 21st century vision of computing; identifies various computing paradigms promising to deliver the vision of computing utilities; defines Cloud computing and provides the architecture for creating market-oriented Clouds by leveraging technologies such as VMs; provides thoughts on market-based resource management strategies that encompass both customer-driven service management and computational risk management to sustain SLA-oriented resource allocation; presents some representative Cloud platforms especially those developed in industries along with our current work towards realizing market-oriented resource allocation of Clouds by leveraging the 3rd generation Aneka enterprise Grid technology; reveals our early thoughts on interconnecting Clouds for dynamically creating an atmospheric computing environment along with pointers to future community research; and concludes with the need for convergence of competing IT paradigms for delivering our 21st century vision.

Recently, many works have targeted satisfying end user requirements using virtualization approach. In this section [12], two of most recognized works are reviewed. Keahey et al. Presented the idea of virtual workspace (VW) which allows users to define an environment in terms of their requirements (such as resource requirements or software configuration), manage and then deploy it in the Grid and Cloud. They have their own Cloud for deployment of VW which named Nimbus Cloud. It provides virtualization in the form of Xen virtual machine and can be used to make a request to deploy a workspace based on a specified VM image. Finally, it has to be mentioned that they have not considered the user QoS requirements and in general SLA. In addition, Cloud discovery and selection is missing from the work. A phenomenal related work has been done in North Carolina State University. The project name is Virtual Computing laboratory (VCL) which was originally described in February 2004. VCL claims that it is an ideal product to support all kind of Cloud solution. VCL services vary from virtual computer laboratory seats or desktops, to single applications on demand, to high-performance computing services and clusters. VCL is now one of the most well-known virtualization management systems in the world, particularly in academia. However, VCL support for images is limited to few types of software and it cannot adequately capture users' varying requirements.

2.3 GridSim: a toolkit for the modeling and simulation of distributed resource management and scheduling for Grid computing

The Grid Modeling and Simulation (GridSim) toolkit provides a comprehensive facility for simulation of application scheduling in different Grid computing environments. However, using the GridSim toolkit to create a Grid simulation model can be a challenging task, especially when the user has no prior experience in using the toolkit before. This paper [2] presents a Java-based Graphical User Interface (GUI) tool called Visual Modeler (VM) which is developed as an additional component on top of the GridSim toolkit. It [8] aims to reduce the learning curve of users and enable fast creation of simulation models. The usefulness of VM is illustrated by a case study on simulating a Grid computing environment similar to that of the World-Wide Grid (WWG) testbed. CloudSim enables seamless modeling, simulation, and experimenting on Cloud computing infrastructures. It is a self-contained platform that can be used to model data centers, service brokers, and scheduling and allocation policies of large scale Cloud platforms. It [7] provides a virtualization engine with extensive features for modeling life-cycle management of virtual machines in a data center, including policies for provisioning of virtual machines to hosts, scheduling of resources of hosts among virtual machines, scheduling of tasks in virtual machines, and modeling of costs incurring in such operations. CloudSim framework is built on top of GridSim toolkit. CloudSim allows simulation of scenarios modeling IaaS, PaaS, and SaaS, because it offers basic components such as Hosts, Virtual Machines, and applications that model the three types of services.

2.4 The MicroGrid: A scientific tool for modeling computational Grids

The complexity and dynamic nature of the Internet (and the emerging Computational Grid) demand that middleware and applications adapt to the changes in configuration and availability of resources. However, to the best of our knowledge there are no simulation tools which support systematic exploration of dynamic Grid software (or Grid resource) behavior. We describe our vision and initial efforts to build tools to meet these needs. Our MicroGrid simulation tools enable Globus applications to be run in arbitrary virtual grid resource environments, enabling broad experimentation.

We [3] describe the design of these tools, and their validation on micro-benchmarks, the NAS parallel benchmarks, and an entire Grid application. These validation experiments shows that the MicroGrid can match actual experiments within a few percent (2% to 4%).There have been many studies using simulation techniques to investigate behavior of large scale distributed systems, as well as tools to support such research. Some of these simulators are GridSim, MicroGrid, GangSim [11], SimGrid and CloudSim. While the first three focus on Grid computing systems, CloudSim is, for the best of our knowledge, the only simulation framework for studying Cloud computing systems. Nevertheless, grid simulators have been used to evaluate costs of executing distributed applications in Cloud infrastructures. GridSim toolkit was developed to address the problem of performance evaluation of real large scaled distributed environments (typically Grid systems but it also supports

IJARSE

simulation of P2P networks) in a repeatable and controlled manner. GridSim toolkit is a Java-based simulation toolkit that supports modeling and simulation of heterogeneous Grid resources and users spread across multiple organizations with their own policies for scheduling applications. It supports multiple application models and provides primitives for creation of application tasks, mapping of tasks to resources, and managing of tasks and resources.

2.5. Scheduling distributed applications: the SimGrid simulation framework

Since the advent of distributed computer systems an active field of research has been the investigation of scheduling strategies for parallel applications. The common approach is to employ scheduling heuristics that approximate an optimal schedule. Unfortunately, it is often impossible to obtain analytical results to compare the efficiency of these heuristics. One possibility is to conducts large numbers of back-to-back experiments on real platforms. While this is possible on tightly-coupled platforms, it is infeasible on modern distributed platforms (i.e. Grids) as it is labor-intensive and does not enable repeatable results. The solution is to resort to simulations. Simulations not only enable repeatable results but also make it possible to explore wide ranges of platform and application scenarios. In this paper we present the SimGrid framework which enables the simulation of distributed applications in distributed computing environments for the specific purpose of developing and evaluating scheduling algorithms. This paper [4] focuses on SimGrid v2, which greatly improves on the first version of the software with more realistic network models and topologies. SimGrid v2 also enables the simulation of distributed scheduling agents, which has become critical for current scheduling research in large-scale platforms. After describing and validating these features, we present a case study by which we demonstrate the usefulness of SimGrid for conducting scheduling research.

2.6 Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities

Cloud computing aims to power the next generation data centers and enables application service providers to lease data center capabilities for deploying applications depending on user QoS (Quality of Service) requirements. Cloud applications have different composition, configuration, and deployment requirements. Quantifying the performance of resource allocation policies and application scheduling algorithms at finer details in Cloud computing environments for different application and service models under varying load, energy performance (power consumption, heat dissipation), and system size is a challenging problem to tackle.

The level on which computing services are offered to consumer varies according to the abstraction level of the service. In the lowest level [5], Infrastructure as a Service (IaaS), services are supplied in the form of hardware where consumers deploy virtual machines, software platforms to support their applications, and the application itself. An example of an IaaS service is Amazon EC2. In the next level, Cloud consumers do not have to handle virtual machines. Instead, a software platform for hosting applications (typically, web applications) is already installed in an infrastructure and offered to consumers. Then, consumers use the platform to develop they specific application. This strategy is known as Platform as a Service (PaaS). Examples of this case are Google App Engine and Aneka. Finally, in Software as a Service (SaaS), an application is offered to consumers, which do not have to handle virtual machines and software platforms that host the application. Reproducible and

IJARSE

controlled experiments on any of these levels require the use of other experimentation methodologies than real execution in a real platform. Simulation is one of such alternatives and this is the focus of this work.

2.7. SimJava: a discrete event simulation library for Java

Simjava is a toolkit for building working models of complex systems. It is based around a discrete event simulation kernel and includes facilities for representing simulation objects as animated icons on screen. Simjava simulation may be incorporated as "live diagrams" into web documents. This paper describes the design, component model, applications and future of simjava.

Recently, many works have targeted satisfying end user requirements using virtualization approach. In this section, two of most recognized works are reviewed. Keahey et al. Presented the idea of virtual workspace (VW) which allows users to define an environment in terms of their requirements (such as resource requirements or software configuration), manage and then deploy it in the Grid and Cloud. They have their own Cloud for deployment of VW which named Nimbus Cloud. It [6] provides virtualization in the form of Xen virtual machine and can be used to make a request to deploy a workspace based on a specified VM image. Finally, it has to be mentioned that they have not considered the user QoS requirements and in general SLA. In addition, Cloud discovery and selection is missing from the work. A phenomenal related work has been done in North Carolina State University. The project name is Virtual Computing laboratory (VCL) which was originally described [10] in February 2004. VCL claims that it is an ideal product to support all kind of Cloud solution. VCL services vary from virtual computer laboratory seats or desktops, to single applications on demand, to high-performance computing services and clusters. VCL is now one of the most well-known virtualization management systems in the world, particularly in academia. However, VCL support for images is limited to few types of software and it cannot adequately capture users' varying requirements.

III. SYSTEM DESIGN

3.1 System Architecture

The Cloud services should be order in the form of queue. The cloud platform is the Graphical user interface acts as the front end controller for the application. The GUI managing screen transitions and other user interface activities.

The cloud storage is the database which is used to store the data entries. Even though Clouds make deployment of large scale applications easier and cheaper, the cloud also creates new issues for developers. Because Cloud infrastructures are distributed, applications can be deployed in different geographic locations, and the chosen distribution of the application impacts its performance for users that are far from the data center. The project proposed an effective architecture to provide construction of virtual server and provide the QOS (performance evaluation) on Cloud service providers

IJARSE

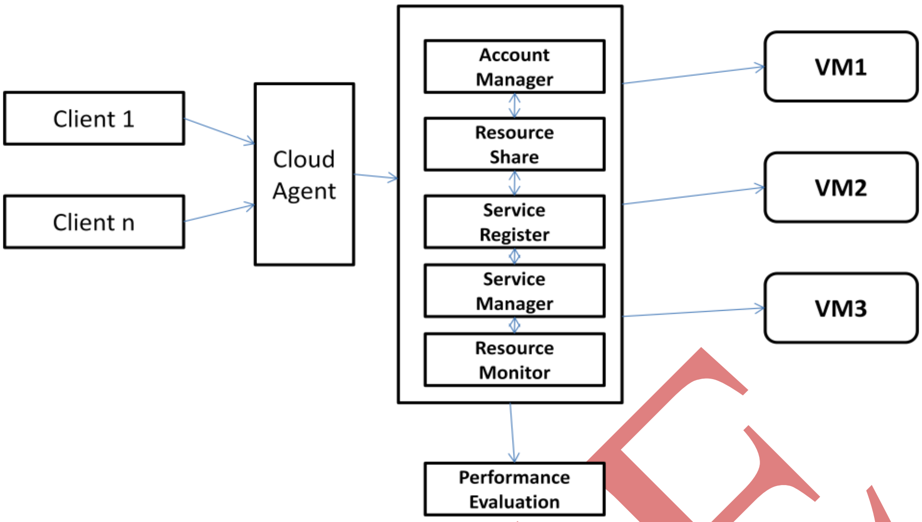

Fig.1 Cloud Service Environment

1. Cloud Agent
2. Account Manager
3. Resource Sharer
4. Service Register
5. Service Manager
6. Resource Monitor
7. Performance Evaluation

3.1.1 Cloud Agent

Cloud Agent is the customers to securely access information from their cloud environment. Leveraging iPhones WiFi/3G and iPod touchs WiFi connectivity, users can securely access corporate reports, network file systems, applications, and content from anywhere, at anytime.

3.1.2 Account Manager

The role of the Cloud Account Manager will be to drive business results and growth by working closely with the assigned cloud partner to establish key application and operator partnerships that lead to increased revenue through the mobile cloud while leveraging the unique capabilities of the product suite.

3.1.3 Resource Sharer

This component provides desirable execution environment based on user requirements and providing necessary disk images and required information for running the application on the IaaS service provider side. After deployment phase, the component helps end users to manage their appliances (for example start or stop them).

3.1.4 Service Register

It allows IaaS providers to advertise their virtual units. The advertisement of virtual unit contains descriptions of their features, costs, and the validity time of the advertisement. From standardization perspective, a common met model that describes IaaS providers' services has to be created.

3.1.5 Service Manager

A monitoring system is provided by this component for fairly determining to which extent an SLA is achieved as well as facilitating a procedure taken by a user to receive compensation when the SLA is violated. The monitoring is based on the copy of signed SLA which is kept in SLA repository. Third party monitoring results can be similar to what the Cloud Status service reports. Hyperic's Cloud Status is the first service to provide an independent view into the health and performance of the most popular Cloud services, including Amazon Web services and Google App Engine.

3.1.6 Resource Monitor

Administrators can drill down into detailed performance data from disk latency to network traffic to application-specific parameters, says the company [2]. Central to Logic Monitor's cloud monitoring is its Active Discovery engine, which provides ongoing discovery of newly added or deleted instances as they are provisioned, and automatically configures them for monitoring. Thousands of virtual devices can be added to monitoring without a need to maintain in-house monitoring servers, the company adds. Logic Monitor is available on a month-to-month subscription basis, which includes maintenance and support.

3.1.7 Performance Evaluation

The performance of the cloud environment is being evaluated using some algorithm and the techniques specified and some optimizing techniques is also specified in the cloud environment.

IV. MODULES

1. GUI package
2. Remote Cloud Server Access
3. Virtual Server Integration
4. Multiple Operating System Integration
5. Security Through Secure Sockets Layer

4.1 GUI Package

The interaction between the requester and the provider takes place via GUI. It provides authentication for the requester and acts as a service provider from the provider. Apart from the cloud authentication the client has to be authenticated for the cloud servers. To be an authentication client, the client has to produce the system name, ip address and port address, and it also specifies the authentication using the username and password, It is stored

IJARSE

in the MySQL database using the WAMP server we can access the database and can perform the import operations. The unauthenticated users can't access the cloud environment.

4.2 Remote Cloud Server Access

The remote Cloud Server Access can be obtained from the remote servers which hold the softwares.The remote servers can be accessed by connecting to the servers through the server ip address and server passwords. And we can access the software from the server by proper authentication environment. The softwares which are not available in our system can be got from the server through the IP configuration.

4.3 Virtual Server Integration

The virtual servers can be integrated through the remote servers.Through the virtual server integration the various software and operating systems can be accessed from the servers. The virtual servers can be accessed from the VMware integrator. And in this module we can calculate the CPU and the exeution time can also be calculated.

4.4 Multi operating System Integration

Through the remote virtual server integration we can access the other softwares.Through our virtual servers we can access the other operating system. For eg., if the client has Microsoft xp in their system but if they want to access Linux that is possible through our virtual server.

4.5 Security through Secure Sockets Layer

The Secure Sockets Layer (SSL) is a commonly-used protocol for managing the security of a message transmission on the Internet. SSL has recently been succeeded by Transport Layer Security (TLS), which is based on SSL. SSL uses a program layer located between the Internet's Hypertext Transfer Protocol (HTTP) and Transport Control Protocol (TCP) layers. The "sockets" part of the term refers to the sockets method of passing data back and forth between a client and a server program in a network or between program layers in the same computer. SSL uses the public-and-private key encryption system from RSA, which also includes the use of a digital certificate. TLS and SSL are an integral part of most Web browsers (clients) and Web servers. If a Web site is on a server that supports SSL, SSL can be enabled and specific Web pages can be identified as requiring SSL access. TLS and SSL are not interoperable. However, a message sent with TLS can be handled by a client that handles SSL but not TLS.

V. CONCLUSION

This work is the first attempt towards developing a tool and an approach for studying large scale distributed applications behavior by simulation in Cloud computing environments. Therefore, the tool will evolve over the time, and the result of this process will improve quality of the model and of the analysis it supports. In the long term this type of experiment has a big potential to aid testers to identify new features and issues, model them,

IJARSE

and develop and evaluate new mechanisms and algorithms for resource management, this way improving performance of emerging Cloud applications.

REFERENCES

[1] BhathiyaWickremasinghe, Rodrigo N.Calheiros and Rajkumar Buyya, "Cloud Analyst: A Cloud sim-based Visual Modeller for Analysing Cloud Computing Environments and Applications", Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications, 2010.

[2] R. Buyya, and M. Murshed, "GridSim: a toolkit for the modeling and simulation of distributed resource management and scheduling for Grid computing", Concurrency and Computation: Practice and Experience, 14(13): 1175-1220, Nov. 2002.

[3] L. X. Song H, Jakobsen D, Bhagwan R, Zhang X, Taura K, A. Chien, "The MicroGrid: A scientific tool for modeling computational Grids", Proc. of the ACM/IEEE Supercomputing Conference, IEEE Computer Society, Nov. 2001.

[4] A. Legrand, L. Marchal, and H. Casanova, "Scheduling distributed applications: the SimGrid simulation framework", Proc. of the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 07), May 2001, pp. 138-145.

[5] R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities", Proc. of the 7th High Performance Computing and Simulation Conference (HPCS 09), IEEE Computer Society, June 2009.

[6] F. Howell and R. Macnab, "SimJava: a discrete event simulation library for Java", Proc. of the 1st International Conference on Web based Modeling and Simulation, SCS, Jan. 2008.

[7] J. Gustedt, E. Jeannot, and Martin Quinson, "Experimental methodologies for large-scale systems: a survey", Parallel Processing Letters, vol. 19, Sep. 2009, pp. 399-418.

[8] E. Deelman, G. Singh, M. Livny, B. Berriman, and J. Good, "The cost of doing science on the Cloud: the Montage example", Proc. of the 2008 ACM/IEEE Conference on Supercomputing, IEEE, Nov. 2008.

[9] M. Assuncao, A. di Costanzo, and R. Buyya, "Evaluating the Cost-Benefit of Using Cloud Computing to Extend the Capacity of Clusters", Proc. of the 18th International Symposium on High Performance Distributed Computing, ACM Press, June 2009.

IJARSE

[10] C. Vecchiola, S. Pandey, and R. Buyya, "High-Performance Cloud Computing: A View of Scientific Applications", Proc. Of the 10th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN 2009), Kaohsiung, Taiwan, Dec. 2009.

[11] C. Dumitrescu, and I. Foster. "GangSim: a simulator for grid scheduling studies", Proc. of the 5th International Symposium on Cluster Computing and the Grid (CCGrid 05), IEEE Computer Society, May 2005.

[12] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities", Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications (HPCC 2008, IEEE CS Press, Los Alamitos, CA, USA), Sept. 25-27, 2008, Dalian, China.

IJARSE