Tools for modelling and simulation cloud computing infrastructure

Ilarion Ishkulov
Saint-Petersburg State University
Saint-Petersburg, Russia
e-mail: ishkulov@gmail.com

Alexander Degtyarev
Saint-Petersburg State University
Saint-Petersburg, Russia
e-mail: deg@csa.ru

Ivan Gankevich
Saint-Petersburg State University
Saint-Petersburg, Russia
e-mail: igankevich@ya.ru

Vladimir Gaiduchok
Saint-Petersburg State University
Saint-Petersburg, Russia
e-mail: gvladimir@gmail.com

Yury Tipikin
Saint-Petersburg State University
Saint-Petersburg, Russia
e-mail: yuriitipikin@gmail.com

Vladimir Korkhov
Saint-Petersburg State University
Saint-Petersburg, Russia
e-mail: vkorkhov@csa.ru

ABSTRACT

Cloud computing is a model of provisioning configurable computing resources, IT infrastructures and applications which can be easily allocated and deallocated by consumer without provider interaction. It can be hard to evaluate performance of newly developed cloud application or infrastructure. Using testbeds for this limits experiments to the scale of the testbed. And achieving reproducible results can be hard or impossible in that case. It's preferable to use simulation tools. Several cloud modelling and simulation frameworks were developed. CloudSim is one of the most powerful. Data centers, physical servers, virtual machines, and applications can be modeled with CloudSim. Application running costs, SLA violations, power usage can be evaluated based on simulated models. In this paper we demostrate feasibility to model real infrastructure with CloudSim framework. For this purpose we model computer infrastructure of Computer Center of St. Petersburg State University.

Keywords

Cloud computing, simulation, CloudSim

1. INTRODUCTION

Cloud computing is widely used for providing and using computing resources which can be easily accessed over a standard network. Cloud based systems try to use resources more effectively by dynamically relocating resources for the most critical tasks. In cloud computing three types of deployment are defined. Public cloud resources available to general public. Private cloud operated solely for a single organization. Hybrid cloud is a composition of two previous models. Private cloud can be free for use inside company but usually limited in resources. Scaling them up introduces new financial cost and can be impossible due to lack of physical space or other limits.

Testing cloud infrastructure is becoming a hot research topic in cloud computing and software engineering community [1]. Deploying large scale infrastructure or application in cloud introduce risks. Testing developed system before deployment can help to find errors and avoid failures in future. Another reason for modelling cloud computing infrastructure is the necessity of predicting QoS for developing application.

Resources from public clouds cost money for their usage. Testing application in public cloud can be cost ineffective. Also other users using public cloud cause testing environment to change in time and tests will be unrepeatable. Private clouds are better for testing. They can be controlled by researchers and may be free for them. But testing large application in private cloud can interfere with other users and critical processes running in the cloud.

Specially build testbeds can be used for testing cloud infrastructure on real hardware. But building real cloud testbed is expensive initiative. Usually such testbeds are limited to small cluster in one geographical location. Small cluster can't provide environment to test applications with large amounts of resources involved. Already deployed testbeds like Open Cirrus [6] project can be used. It has own policies for allowed projects can be non-free and introduce same problems users meet while working with common public cloud provider like changing environment time as other users may use resources at the same time which decrease test repeatability.

In order to provide controllable and repeatable tests that can be run on a researcher's existing hardware, a purpose built simulator can be created. This is done by abstracting some aspect of the system that the researcher simulates, creating a model for its behavior and then implementing the model in a programming language. The advantage of this approach is that the resulting simulation will be small and efficient, requiring minimal computing resources. The disadvantage is that any model created will need to be validated to ensure the results are within an acceptable error margin. Additionally, if the system being modeled changes significantly or the researcher desires to model additional features of the system or interactions between systems, the model will need to be changed or rewritten entirety.

2. CLOUD SIMULATORS OVERVIEW

Several simulators have been specifically developed for performance analysis of cloud computing environments

[1], including CloudSim, GreenSim, SPECI, GROUDSIM, and DCSim.

2.1 GreenCloud

GreenCloud is a packet-level simulator for energy-aware cloud computing data centers with a focus on cloud communications. It offers a detailed fine-grained modeling of the energy consumed by the data center IT equipment, such as computing servers, network switches, and communication links [3].

2.2 SPECI

Simulation Program for Elastic Cloud Infrastructures (SPECI) allows analyzing large data center behavior under the size and design policy of the middleware as inputs [4].

2.3 DCSim

Data Center Simulator is concentrated on modeling data centers, offering IaaS to Multiple tenants, in order to achieve a simulator to evaluate and develop data center management techniques [5].

3. CLOUDSIM ARCHITECTURE

CloudSim is a leading technology in simulating cloud computing infrastructure. CloudSim allows to carry out discrete event simulation (DES) of cloud computing infrastructure [2]. CloudSim can model large scale Cloud computing data centers on a single physical node. It's open source software. First version was released in year 2009.

CloudSim can model different virtualized infrastructures: cloud based data centers, virtualized clusters, public clouds, private clouds, hybrid clouds. With its functionality for virtualized infrastructures simulating it can be used to model smaller virtualization clusters. CloudSim exposes functionalities for complex workload profiling and application performance studies. Testing and profiling application on models can be done while cluster is used by other users.

CloudSim has a multi-layered design, shown in Figure 1. The initial version of CloudSim was based on JavaSim. However in later versions it has been replaced by newly developed discrete event management (CloudSim core simulation engine). It was done for introducing such features as resetting simulation programmatically at a run time, creation of new simulation entities at run time and performance improvement.

The CloudSim simulation layer provides functions for modelling virtualized Cloud-based data center. It controls simulation of such things as virtual machines (VMs), hosts data centers, applications. Different policies in allocating hosts can be implemented by programmatically extending the core VM provisioning functionality. User Code layer provides the functionality for configuring hosts (number of machines, memory, storage, bandwidth and so on), applications, VMs, number of users and their application types, and broker scheduling policies.

Datacenter entity manages number of host entities. Host represents a physical computing server. CloudSim support simulating single-core and multi-core nodes. Host has preconfigured processing capabilities (in MIPS), mem-

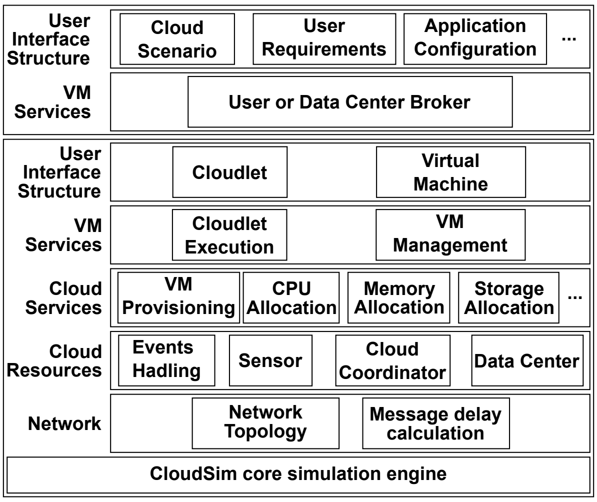

Figure 1. CloudSim layered architecture

ory, storage and provisioning policy for allocating processing cores to virtual machines.

CloudSim has good support for modeling virtualization enabled resources. It supports VM provisioning in two layers: at host level and at the VM level. At host level amount of processing power of each core assigned to the VM can be configured. At the VM level amount of processing power can be assigned to individual application services.

In public clouds pay-as-you-go model is adopted for resource trading. CloudSim provides functionalities to model costs and economic policies. Cost per unit of memory, cost per unit of storage, cost per unit of usage of used bandwidth for IaaS layer services. At the SaaS layer costs are evaluated based on application service requests.

CloudSim uses latency matrix for calculation time that takes message to travel from one CloudSim entity to another. The latency matrix is generated at initialization time from topology description stored in BRITE format. For better modeling network behavior RealCloudSim can be used. It is based on CloudSim main engine but has network simulation based on NS2 (Network Simulator 2).

CloudSim supports modeling and simulation of Cloud computing environments consisting of inter-networked clouds (federation of clouds). Different policies and provision techniques can be modelled for evaluation internetworked Cloud computing scenarios. Hybrid cloud infrastructure can be modeled for evaluation of benefits from expanding private cloud to hybrid cloud.

CloudSim support modeling varying application requirements for computational resources. Important requirement for cloud is to ensure SLA in terms of QoS parameters such as availability, reliability and throughput are delivered to the applications. ClousSim supports modeling SLA violation scenarios. The number of SLA violation events as well as the amount of resources that was requested but not allocated can be accounted for by CloudSim. That can be used to evaluate cloud application behaviours when they try to use all of the available resources or when resources are limited.

ClouSim framework provides basic models to validate and evaluate power consumption of cloud infrastructure.

Different power consumption models can be simulated such as Dynamic Voltage and Frequency Scaling.

Resources in cloud can be easily provisioned and deprovisioned. Actual usage patterns of many enterprise services (business applications) vary with time, most of the time in an unpredictable way. This leads to the necessity for Cloud providers to deal with customers who can enter or leave the system at any time. CloudSim supports dynamic creation and adding new entities to the running simulation such as VMs, Host and even Data Centers. That gives functionality to test dynamic environments where system components can join or leave the system.

4. CLOUDSIM IMPLEMENTATION

Each CloudSim entity implemented as java classes.

Datacenter: Encapsulates host allocations. Implements policies for allocating bandwidth, memory and storage devices to hosts and VMs.

Host: Models physical servers. It stores server configuration such as memory, storage and bandwidth and policies used for allocating then to VMs running on the host. Processing power defined as Pe class which represents CPUS/Cores. And stores provision policy for VMs like time-shared or space-shared.

DatacenterBroker: This class models a broker, which is responsible for mediating negotiations between SaaS and Cloud providers; and such negotiations are driven by QoS requirements. The broker acts on behalf of SaaS providers. It discovers suitable Cloud service providers by querying the Cloud Information Service (CIS) and undertakes on-line negotiations for allocation of resources or services that can meet applications QoS needs. The researchers and system developers must extend this class for evaluating and testing custom brokering policies. The difference between the broker and the CloudCoordinator is that the former represents the customer (i.e., decisions of these components are made in order to increase user-related performance metrics), while the latter acts on behalf of the data center, i.e., it tries to maximize the overall performance of the data center, without considering the needs of specific customers.

Vm: This class models virtual machine. VMs are managed by Host class. It stores characteristics of available resource: memory, storage, etc. And define internal VM's provisioning policy implemented as extended CloudletSheduler component.

Cloudlet: represents application process running on virtual machine. It defines int terms instructions to be executed and amount of data transfer to complete the task. It encapsulates UtilizationModel. Method getUtilization() of class UtilizationModel can be overridden to described resources usage of cloudlet based on time.

5. RELATED PROJECTS

5.1 RealCloudSim

RealCloudSim uses CloudSim core engine. It provides graphical interface to read network topologies. It simulates network with NS2 (Network Simulator 2). It has own engine for allocation simulation based on Genetic Algorithms.

5.2 CloudAnalyst

Cloud Analyst is a tool developed at the University of Melbourne whose goal is to support evaluation of social networks tools according to geographic distribution of users and data centers. In this tool, communities of users and data centers supporting the social networks are characterized and, based on their location; parameters such as user experience while using the social network application and load on the data center are obtained/logged.

6. DEVELOPING A MODEL USING CLOUDSIM

As an experiment we developed a model of real cluster and tried to compare metrics we can get from model and from real hardware. Resource Center Computation Center of St. Petersburg State University was used as real hardware test bed. It adopts virtual space approach that borrows principles of building private-clouds [7].

Cluster is built with HP blade servers with 2 Intel Xeon X5670, 96GB RAM, interconnected with 10GbE and QDR InfiniBand. As a storage HP StorageWorks P4500 G2 is used with 120 2TB SAS hard drives. VMware hypervisor is used for virtual infrastructure.

We used dummy applications that need X CPUs on H different VMs. They consume M amount of RAM, B of bandwidth, S of storage. X, H, M, B and S are parameters that can be changed by experimenter. Application allocates specified amount of memory, writes file to disc and send data to the other host using resources of the virtual machine. Applications were simple enough so that they could be modelled by simple CloudSim Cloudlets.

Comparing output of metrics monitored on the real hardware during execution of test application and results that were produced by simulation we can see small divergence. But modelled application behaviour shows the general trend of execution. That shows us possibility to predict applications behaviours. With models build with CloudSim framework test can be conducted without deployment application to the real infrastructure.

7. ACKNOWLEDGEMENT

Work is partially supported by the Russian Foundation for Basic Research, project no. 13-07-00747-a and St. Petersburg State University, project no. 9.38.674.2013

8. ADDITIONAL AUTHORS

REFERENCES

[1] A. Oujani, "A Survey on Cloud Computing Simulations and Cloud Testing", *http://classes.cec.wustl.edu/~ao7/*

[2] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, Rajkumar Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms", *Software: Practice and Experience, Volume 41, Issue 1*, pp. 23-55, 2011.

[3] D. Kliazovich, P. Bouvry, S. U. Khan, "GreenCloud: a packet-level simulator of energy-aware cloud computing data centers", *IEEE Global Telecommunications Conference*, pp. 1-5, 2010.

[4] I. L. Sriram, D. Cliff, "SPECI-2: An open-source framework for predictive simulation of cloud-scale data-centres" *http://arxiv.org/abs/1106.5465*

[5] Open Cirrus, *http://opencirrus.org/*

[6] A.V. Bogdanov, A.B. Degtyarev, V.Yu. Gaiduchok, I.G. Gankevich, V.I. Zolotarev, "Virtual Workspace as Basis of Supercomputer Center", *Distributed Computing and Grid-Technologies in Science and Education: Proceedings of the 5th Intern. Conf. Dubna: JINR*, pp. 60-65, 2012