ET2539: "Green Internetworking"

LABORATION: CLOUD SIMULATION

Blekinge Institute of Technology
School of Computing
prepared by Selim Ickin, October 2013

1 Introduction

This is the project description that will enable you, as a cloud developer, to apply your Green Networking knowledge into cloud and technically observe the intervals of how it can be turned into benefit in a simulation environment. You will be introduced with a simulation tool called *CloudSim*, that provides you to reserve resources in the cloud and observe the corresponding technical parameters. CloudSim is a toolkit (library) for simulation of Cloud computing scenarios. It provides basic classes for describing data centers, virtual machines, applications, users, computational resources, and policies for management of diverse parts of the system (e.g., scheduling and provisioning). The CloudSim simulation layer provides support for modeling and simulation of virtualized cloud-based data center environments including dedicated management interfaces for Virtual Machine (VM)'s, memory, storage, and bandwidth.
In addition, CloudSim can simulate Infrastructure as a Service (IaaS) by extending the datacenter entity that can manage number of host entities. Entity in CloudSim corresponds to an instance of a component, where a component can be a class, or set of classes that represent one CloudSim model, e.g., data center, host. Each host is assigned to one or more VMs. The VM allocation is defined by the Cloud service provider. For more information on CloudSim, please visit [1].
What can you do with CloudSim?

- generate a mix of workload request distributions, application configurations
- model Cloud availability scenarios and perform robust tests based on the custom configurations
- implement custom application provisioning techniques for clouds and their federation

2 Learning Objectives

After having completed the project you will be able to:

- use the classes to describe data centers, virtual machines, applications, users, computational resources, and policies for management of parts of the system, e.g., scheduling and provisioning.
- evaluate and understand the efficiency of strategies from different perspectives, from cost/profit to speed up the application execution time
- understand how the parameters in different layers of the cloud architecture affect the overall performance
- improve your technical and communication skills within group works.

3 Requirements

Basic knowledge on java programming and cloud computing architecture is required. It's highly recommended that you work on Linux. If you don't use Linux, we encourage you to run a virtual machine on your host operating system and install Ubuntu. Then you will need to install java compiler of version 1.7. In other versions, you might face issues compiling your cloud simulator codes. You can also use the computers in BTH labs (please send an email to selim.ickin@bth.se if you need a remote linux machine). We will not allocate any labs explicitly for the course purposes.

4 Objective

The objective of the project is to help students to understand the basic concepts behind cloud computing. You will be configuring a cloud with a set of parameters such as the number of users, RAM, bandwidth, number of MIPS, image size, number of CPU's, processing / memory / storage / bandwidth costs, etc. You will be studying the influence of those parameters on the total debt of the users to the datacenters, the energy consumption of datacenters, and execution times of the cloudlets. Some metrics under investigation will be the bandwidth and CPU utilization of the data centers, waiting time of the cloudlets executed on a resource, debts of users to the datacenters, energy consumption of datacenters, etc.

5 Procedure

1. Each project group will consist of two students and they will be formed randomly. Make sure you are involved in one group after we announce the project groups in ItsLearning. If you have problems finding a group, please contact us.
2. First step should be to define the needs for developing the application. You only need a basic knowledge on java programming. CloudSim provides you example codes, where you can see detailed descriptions of the codes as well. If you feel weak on java programming the subjects listed in Section 3, there are plenty of resources available on the web,[3].
3. Read the paper as cited in [2].
4. Download and install your development environment. **Hint:** *apt-get install openjdk-7-jdk* would do it for you.
5. Set up a cloud with different configurations, *i.e.*, follow the steps in Section 6.

6 Project Description

6.1 Part A

1. Create 2 datacenters (Datacenter1 and Datacenter2) with the following parameters:
   (a) Architecture: x86
   (b) Operating System: Linux
   (c) Hypervisor: Xen
   (d) Scheduling Interval: 30
   (e) Allocation Policy: Single threshold
   (f) Upper threshold: 0.8 (must be set between 0 and 1)
   (g) Lower threshold: 0.2 (must be set between 0 and 1)
   (h) Enable the VM migrations
2. Add 1 host to each datacenter with the following parameters:
   (a) RAM: 40000
   (b) BW: 10000000

(c) Storage: 1000000

(d) Max Power: 250

(e) Static power percent: 0.7

(f) VM Scheduling: Time shared

(g) Set the power model to linear

(h) MIPS/PE: 2400

(i) Processing elements: 4

(j) Power model: Linear

(k) Set RAM, BW, and PE Provisioners to Simple

3. Set the costs as follows:

(a) Processing cost (per sec): 0.1

(b) Memory cost (per MB): 0.05

(c) Storage cost (per MB): 0.001

(d) Bandwidth cost (per MB): 0.1

4. Set the Storage Area Network (SAN) parameters as follows:

(a) Capacity: 10000000

(b) Bandwidth: 10

(c) Latency: 5

5. Set 4 customers and each customer will have two VMs. Set the following configurations. Customer VM :

(a) Image size: 1000

(b) Proc. Elements: 1

(c) MIPS: 1000

(d) RAM: 512

(e) BW: 100000

(f) Priority: 1

(g) Hypervisor: Xen

(h) Scheduling policy: Dynamic workload

(i) Set the utilization profile as follows:

(j) Broker policy: Round Robin

(k) Processing elements: 1

(l) Maximum length: 50000

(m) File size: 500

(n) Output size: 500

(o) CPU Utilization model: Stochastic

(p) RAM utilization mode: Stochastic

(q) Bandwidth utilization model: Stochastic

6. Make sure the link bandwidth and the link delay between each node are set to 1.
7. Run the simulation
   (a) Observe the simulation time.
   (b) Write the summary of your configuration, i.e., the number of customers, number of hosts, number of datacenters, etc.
   (c) Plot the CPU (MIPS) and RAM (MB) vs. Time graph for the two datacenters. (y-axis: Resources, x-axis: Time).
   (d) Plot the overall Power Consumption vs. Time graph for the two datacenters.
   (e) Observe of the number of successfully allocated virtual machines by the customer for the two datacenters.
   (f) Observe the number of successfully executed cloudlets on each customer for the two datacenters..
   (g) Observe the costs generated by each customer based on their utilization on the datacenter for the two datacenters.
   (h) For each virtual machine (VM), plot the CPU (MIPS) and RAM (MB) vs. Time graph.
   (i) Observe the number of successfully executed cloudlets on each VM.
   (j) Observe the debt of each user for each data center.
   (k) Observe the start times and the finish times of the cloudlets for each resource and virtual machine.

6.2 Part B

1. Add four more customers, i.e., in total there will be 8 customers.
2. Configure each customer to have two VMs with exactly the same settings.
3. Change the utilization costs of Datacenter2 as follows:
   (a) Processing cost: 1
   (b) Memory cost: 0.5
   (c) Storage cost: 0.01
   (d) Bandwidth cost: 1
4. For Customer2 and Customer 5, set the VM RAM to 64 and VM Bandwidth to 50.
5. For Customer5, set the VM RAM to 128 and VM Bandwidth to 100.
6. For Customer8, set the network link to all customers and to all datacenters such that Bandwidth=0.1 and Latency=1000.
7. Run the simulation
   (a) Observe the simulation time.

(b) Write the summary of your configuration, i.e., the number of customers, number of hosts, number of datacenters, etc.

(c) Observe the number of successfully executed cloudlets on each datacenter for each customer.

(d) Observe the costs generated by each customer based on their utilization on the datacenter.

(e) Observe the number of successfully executed cloudlets on each VM.

(f) Observe the debt of each user for each data center.

8. For Customer4, set 10 virtual machines, and rerun the simulation.

   (a) How many VM's are running in each datacenter for each customer?

   (b) Did the cost of Customer4 change?

   (c) Elaborate on the scheduling of cloudlets on the virtual machines of Customer 4 by looking at the start and finish times of the cloudlets. Plot the allocation of each VM by a cloudlet over time on the two datacenters.

7 Deliverables

Each group is expected to submit one project report. The deliverable will be a project report with the name: *labGroupNo_acronym.pdf*. Your final project report should be well structured and should include the chapters namely, *Introduction, State of Art, Methodology, Experiment Settings, Results, Conclusion, and References.* Observe that plagiarism is definitely prohibited. In the Results section, it is expected to include results that are supplemented by plots or tables with respect to the results you obtained as defined in Section Objective. Do NOT use a two-column IEEE format in your report; the template is located under the Projects legend in itsLearning. The results section of the final project report must include a creative way to explain your observations (e.g., differences in results between Part A and Part B). Please, do not include screenshots with unreadable text or redundant information.

8 Examination

There will be certain time slots (to be announced in itsLearning) allocated for each group to present their work. So, stay tuned. Each group member MUST be present with her/his group members during the presentation. The deadline of the project is to be defined. You can always post your questions on itsLearning, and we will try to help you as much as we can. Observe that you can either pass or fail the project, *i.e.*, your grade will be U or G.

References

[1] CloudSim, `http://code.google.com/p/cloudsim/`

[2] Rodrigo N. Calheiros, Rajiv Ranjan, Anton Beloglazov, Csar A. F. De Rose, and Rajkumar Buyya. 2011. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Softw. Pract. Exper. 41, 1 (January 2011), 23-50. DOI=10.1002/spe.995 http://dx.doi.org/10.1002/spe.995.

[3] CloudSim API, `http://www.cloudbus.org/cloudsim/doc/api/index.html`