Specialist English: Assignment 7

Rebecca J. Stones
rebecca.stones82@nbjl.nankai.edu.cn

Date due: 7 December 2017

In this seventh assignment (worth 5% of the final mark), we will look at the introductory material in a computer science paper.

I'll scale the marks on this assignment according to $m \mapsto \min(m, 10)$ for Master's students and $m \mapsto \lceil m/1.3 \rceil$ for Ph.D. students.

My marking will be affected by (a) your English writing, (b) your LaTeX typesetting, (c) your mathematical presentation, and (d) your understanding of the underlying computer science. Basically, I will "peer review" your assignments.

Problem 1 The following snippet is a typical example of a topic sentence and a conclusion sentence in the first paragraph of the Introduction. Describe what you expect will be between these two sentences. [1 mark]

> Community Question Answering (CQA) websites have become valuable knowledge repositories in their specific domains (Pudipeddi et al. 2014; Droretal. 2012; Nie et al. 2015).
>
> ⋮
>
> As a successful example, Stack Overflow, a CQA website specializing in programming, has attracted 5,277,831 registered users and the number is still rapidly increasing.
>
> — Liu et al., *Toward Personalized ...*, TOMM (2018).

The remainder of the questions in this assignment relate to the following paper (and I attach the first two pages):

> Wang et al., *CloudEval: A Simulation Environment for Evaluating the Dynamic Cloud VM consolidation*, SIMUTOOLS (2016).
>
> Available from: `https://dl.acm.org/citation.cfm?id=3021433`

(Note that Wang et al. (2016) has many grammar and spelling errors.)

Problem 2 In Wang et al. (2016), identify the topic sentence and the conclusion sentence in the first paragraph of the Introduction. [1 mark]

Problem 3 In Wang et al. (2016), identify the "table of contents" paragraph in the Introduction. [1 mark]

Problem 4 In Wang et al. (2016), writing "more and more" is somewhat informal: what is a suitable alternative? [1 mark]

Problem 5 Rewrite the "major contributions" snippet to make it grammatically correct and flow naturally (it begins "To further ..." and ends "... of cloud simulation"). I recommend looking at some high-quality papers to get a feel for how it is ordinarily written. [4 marks]

The answer to Problem 5 needs to address the following issues:

1. the noun "purpose" is singular, yet three items are listed;

2. the word “aspects” is misused (in fact, I don’t recall ever seeing “aspect” used correctly in publications by Chinese authors; I think it’s translated from 方面, and in particular 以下方面);

3. the sentences beginning “Design a ...” and “Simplify the ...” do not have a subject;

4. “Domain-Specific Language” is not a proper noun (and thus should not be capitalized);

5. “the process of algorithm implementation” is clunky (here “implementation” is the “process”); similarly “do a comprehensive survey” is clunky (here we can use “survey” as a verb to describe what we “do”);

6. the word “do” is what children say (because they struggle to be more precise);

7. the acronym “VMs” is incorrectly used as an adjective while written as a plural noun; and

8. “knowledge” is not the correct word: “expertise” is, but we need to adjust the grammar to make this replacement.

Problem 6 In Wang et al. (2016), the Related Work section describes “CloudSim”, “NetworkCloudSim”, and “Groovy”. How important are these programs to the relevant paper, and how much text is spent describing them? [1 mark]

Problem 7 Do the authors describe how the proposed method differs from “CloudSim”? If so, where? [1 mark]

Problem 8 What do we learn about “Green Cloud” in the Related Work section? [1 mark]

Problem 9 Compare Figure 1 in Wang et al. (2016) with Figure 3 by Calheiros et al. (2011). What is a major problem here? What is a potential consequence of this problem[1]? How could we easily avoid this problem? [2 marks]

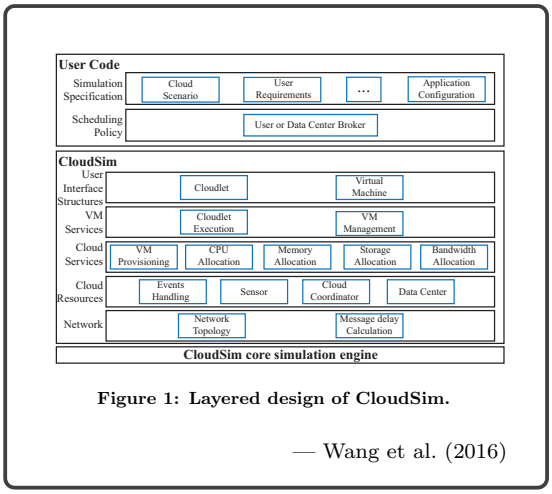

Figure 1: Layered design of CloudSim.

— Wang et al. (2016)

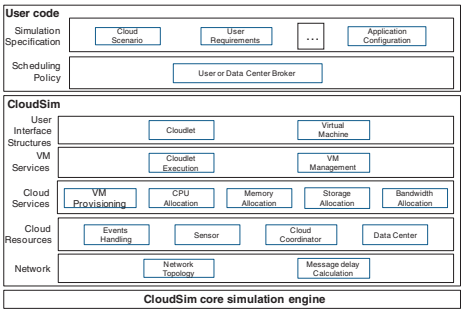

Figure 3. Layered CloudSim architecture.

— Calheiros et al., *CloudSim: a toolkit for modeling ...*, Softw. Pract. Exper. (2011)

[1]It would help to visit ACM’s Plagiarism Policy webpage: `https://www.acm.org/publications/policies/plagiarism`

CloudEval: A Simulation Environment for Evaluating the Dynamic Cloud VM consolidation

Zhi Wang
Software College
Northeastern University China
Shenyang, Liaoning, China
yfwz100@163.com

Zhaohua Zhou
Software College
Northeastern University China
Shenyang, Liaoning, China
zhou.zh@qq.com

Shilin Jia
Software College
Northeastern University China
Shenyang, Liaoning, China
958240037@qq.com

Hao Liu
Software College
Northeastern University China
Shenyang, Liaoning, China
18202409257@163.com

Dancheng Li*
Software College
Northeastern University China
Shenyang, Liaoning, China
*Corresponding Author
Ldc@mail.neu.edu.cn

Jingde Cheng
Department of Information and Computer Sciences
Saitama University
Saitama-shi 338-8570, Japan
cheng@ics.saitama-u.ac.jp

ABSTRACT

Virtual machine(VM) scheduling algorithms optimize the energy consumption without performance degradation, where cloud providers profited greatly. However, not any algorithms performs the same. To evaluate the power efficiency of such algorithms, various scale of cloud clusters have to be maintained, which is costly and difficult. In this paper, we present a emulation software *CloudEval* to evalute the performance of migration algorithms of VMs. Comparing to previous cloud simulation environment, we addresses the evaluation of dynamic scheduling of the VMs in the cloud, by a simpler and a non-intrusive manner. Most previous cloud simulation environments encapsulate the simulation process in a close form(eg. the coupling of simulation and evaluation), which is not easy for researchers to design a VM scheduling algorithm and perform the evaluation. Besides, it suggests the clear test cases of workload and energy model, which is also important in VM scheduling algorithm evaluation.

Specifically, we address 3 aspects in the evaluation environment: the open architecture of simulation framework for algorithm/strategy design, the evaluation metrics, the simulation cases of workload and mapping of realistic cloud environment. The framework is intended to provide references for algorithm researches.

CCS Concepts

•Computing methodologies → Simulation environments; •Computer systems organization → *Cloud computing;*

SIMUTOOLS 2016, August 22-23, Prague, Czech Republic

Keywords

cloud computing, virtual machine live migration, performance evaluation tools

1. INTRODUCTION

The fast developing economics in cloud computing is gaining more and more attention for the energy profiling of the data center. Nowadays, as the virtualization engaged in most cloud infrastructure, the scale of data center is much larger than before. However, much of the servers are underutilized [10] [20] [16]. With the development of live migration technology of VMs [5], dynamically managing the layout of the VMs becomes possible. The dynamic scheduling of VMs is to dynamically allocate/consolidate VMs to achieve better trade-off between performance and energy consumption. Various energy conservation alogirhtms for the management of VMs is thus developped since then and proposed a concept of "Green Computing" [20]. However, by utilizing the energy-saving algorithms, the performance of the data center may be degraded by the dynamic consolidation of the VMs. One obvious reason is that VM has to be paused when migrating from the original host. Besides, the number of VM allocated in a physical host also affects the performance of the host, where VMs tend to compete for the limited resources. Notably, most algorithms tend to make the host overloaded by putting as many VMs as possible, which by turn results in a performance degradation when there's possible peak in resource utilization [16].

Evaluating the algorithm of VM dynamic scheduling requires theoretic analysis and experiments. However, setting up a large scale of machine clusters is not practical to many researchers. Basically proposed methods are evaluated in a self-constructed non-standard environment on simulation software, like matlab. It is a cheap way to verify the effectiveness of the algorithms. However, there's the risk to misleading the researcher because the complexity of the simulation environment. The scheduling of the VMs involves the scheduler of VM supervisors(on physical hosts) and the scheduler of the cloudlets or programs. Literatures make assumptions on the simplification on the simulation envi-

ronment and few of them have done the experiments on the same condition. Li et al. make the first attampt to do a comprehensive evaluation of scheduling methods of VM migration for energy conservation [11]. However, the implementation detail of the experiment environemnt is omitted.

In this paper, we introduce a new emulation software to perform the evaluation on VM consolidation, based on the famous cloud infrastructure simulation software CloudSim [4]. To further describe the work, purpose of our work is addressed in the following aspects:

1. Design a Domain-Specific Language to better facilitate the process of algorithm implementation and experimental automation. To achieve this goal, we do a comprehensive survey about how authors design their VM migration strategy to carefully design the interfaces.

2. Design and maintain a consistent simulation environment for evaluating energy-efficient VMs dynamic scheduling algorithms.

3. Simplify the design of evaluation metrics for researchers who attempt to evaluate dynamic scheduling algorithms of VMs, without the knowledge of cloud simulation.

In Section 2, we will survey about simulation of cloud infrastructures. Section 3 will introduce the architecture of the cloud evaluation and the common simulation loop. Section 4 discusses the issues in simulation of the cloud and proposes the potential fixes. Later Section 5 gives a case study to evaluate a VM scheduling algorithms. Finally, Section 6 discusses the defects and suggests the future work. Finally, Section 7 gives a conclusion.

2. RELATED WORK

As CloudEval is an evaluation framework based on simulation environment, the related works will includes basis of the simulation environment, the specific-designed language as well as the foundation of the evaluation process, which build up the foundation of the framework.

As we survey about the simulation environmnt, the research of the cloud infrastructure is initiated in about 2009 closely followed by the concept of Cloud Computing suggested by Google in 2006. In the past decades, several simulation environment is proposed. Among these simulation tools, CloudSim[4] is the most comprehensive simulation environment for cloud computing at the present. After its release, many researchers extended CloudSim to many other aspects in order to do the related experiments, like NetworkCloudSim[9]. At the same time, the concept of Green Cloud is suggested. Many researchers try to optimize the cost and the energy consumption and CloudSim is extended to such fields[2]. However, many researchers just extended CloudSim to a special case and haven't do experiments on the same condition[11]. Besides, CloudSim builds around the entity and relation from the cloud infrastructure, as shown in Figure 1 . It doesn't provide guidelines to facilitate the development of evaluation metrics designs and algorithms implementation. It lacks the standard interfaces to monitor the data and formulate the evaluation reports. Our work will base on the cloud simulation concept of CloudSim, with emphasis on parts of the performance evaluation of VMs migration.

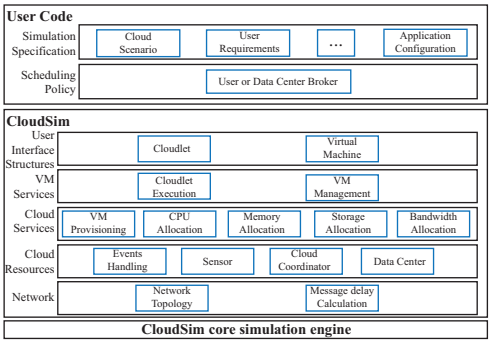

Figure 1: Layered design of CloudSim.

To distinguish from the CloudSim, we extract the important entities from the CloudSim and make our own engine and domain specific language, as shown in Figure 2. The contribution of CloudSim is the clear separation of cloud levels and the interactions of entities between each level. However, in an evaluation process, the interactions between entities is not enough. Our work will focus on the evaluation process, which is the upper layer of the CloudSim.

To the best of our knowlege, this is the first attampt to build a specifically-designed language for evaluation on CloudSim or similar simulation environment. Since CloudSim is built upon Java. In order to reuse the code, a Java-based language is selected as a basis of the design. Groovy[18] is a general-purpose programming language based on JVM with the thoughts of domain specific language design in its core. It can delegate the running context to create custom control structures and operator overloadings to build the symbolic programming[17]. With the flexibility of the syntax, a specific designed of the evaluation can be built clearly on the CloudSim and our evaluation metrics.

For evaluation metrics defined to build the evaluation process, we surveyed about many literatures[2][11][13][12]. [11] is the first attampt to sum up a comprehensive evaluation process and metrics in VM migration evaluation. It sums up most of the metrics seen in the literatures before and evaluate many algorithms at the same simulation platform. This is the attampt to build a flexible and solid foundation of the evaluation process. Therefore, as a simulation framework, we will focus on the validation of the process and the ability to extend the evaluation metrics. The validation of the simulation process is achieved by the base of CloudSim and the extension which defines the flexible interfaces to build evaluation metrics refered mostly in [11] and other works.

3. ARCHITECTURE OF THE SIMULATION

Firstly, we will demostrate the overall layer design of the architecture and how we build the simulation process with the detailed differences from CloudSim. As shown in related works, the layered design of CloudSim provides a clear seperation of cloud infrastructures. As the evaluation process of the VMs performs on migration, it mainly focus on the "Infrastructure as a Service"(IaaS) layer of the overall cloud framework, which is the basis of the "Plaftorm as a Service"(PaaS) and "Software as a Service"(SaaS) layer. In