Generating fluctuating workload for cloud elasticity simulations

Simon Bihel (Student) Computer Science Department, ENS Rennes

simon.bihel@ens-rennes.fr

***Abstract*—Cloud computing is a model that makes available infrastructures, platforms and software with a pay-as-you-go subscription. It aims to reduce the cost with a layer of visualization that allows virtual resources to be dynamically adjusted and occupied on-demand. The problem of using the minimal resources for the current demand/usage is still a research challenge that spans all layers and applications. This dynamic management of clouds is called cloud elasticity.**

***Index Terms*—Simulation; Cloud elasticity; Workload**

I. Introduction

II. Background

The global architecture of a cloud is described in the Figure 1. It is split in different layers and each layer has a specific role in the business model of clouds. On the lowest part there is the physical resources (e.g. data centers) with Infrastructure as a Service model (IaaS). The pricing is based of the resources available. Then comes the layer of virtualization and performances negotiation called Platform as a Service (PaaS). Dealing with Virtual Machines (VMs also called hosts) allows a cleaner sharing of resources and makes it easier to answer the users' demands (e.g. deploying more VMs). The pricing depends of multiple factors, like the Quality of Service (QoS) for the quality of the network, the Service Level Agreement (SLA) for the faults rate, handling and responsibilities... On top of that is the layer for users' cloud tools with Software as a Service (SaaS). These tools will allow the user that writes cloud applications to manage resources, run their code...

Most cloud applications will have fluctuating workload over time. For example with a website server the usage might be bigger during the day or during a short period because of a viral cultural event. Resources should thus be managed dynamically. Also, physical resources can encounter problems which makes them unavailable. Because of all these constraints the SLAs and QoS negotiations aren't satisfied easily and 100% availability is never a thing. On top of that every actor wants to meet the obligations with a minimal cost. All these questions of availability and cost are current research problems. In particular we are interested in the works that tackle problems related to fluctuating and dynamic usage of cloud applications and resources. The ability for a cloud infrastructure to adapt to a dynamic workload is called cloud elasticity.

There are some generic elastic actions. The act of deploying more VMs (and thus having more resources overall) is called scaling up (and the convert is scaling down). The act of moving VMs to a different location is called scaling out. This is used for example when time passes by and users come from different countries/continents.

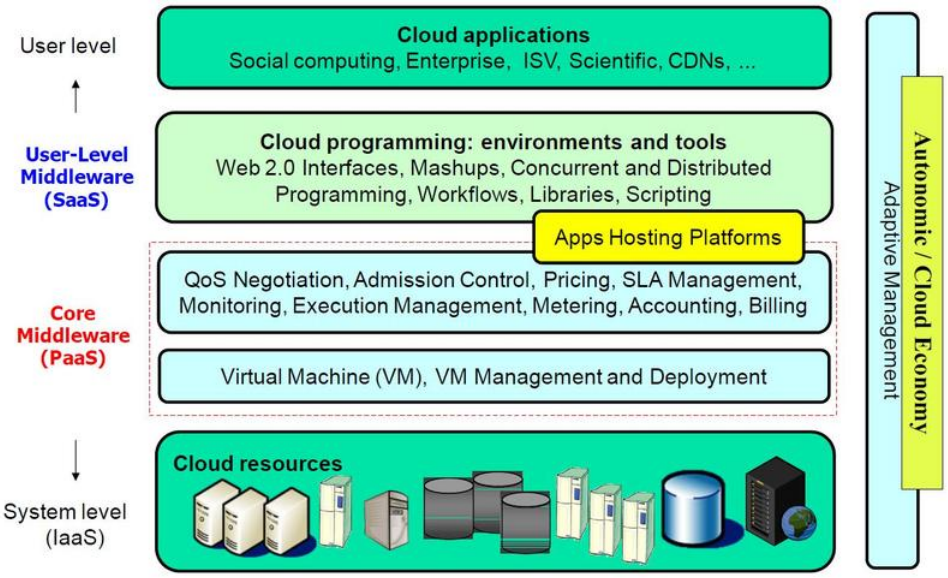
[1] has categorized works on cloud elasticity and allows to see which elements of a cloud infrastructure, platform or application/software are impacted. As it is for now most research works are evaluated on real clouds It is interesting for a distributed systems simulator to search what is needed for simulating cloud elasticity. If it is shown that research works on cloud elasticity can be evaluated on a simulator they would benefit from cost reduction, re-runable experiments, trust in results...

In this survey proposals are categorized as follows. The scope is about what elements of a cloud the proposals work on. It can be the management of VMs, allocation of resources... Then there is the purpose of the proposal. Enhancing the *performances* (to meet the SLA), reducing the *energy consumption* footprint, being *available* when needed and reducing the overall *cost*. Another dimension is the decision making. This is what a proposal add to an existing cloud to pursue its purpose. In addition to the scope there is the elastic actions performed by the proposals. As the scope is about what elements of a cloud are concerned, the elastic action is about what is done to them. Then there is the provider dimension that tells if there is only one provider or multiple ones. At last there is the method used by the proposal to evaluate itself, through real cloud, simulation or emulation.

The survey gives a good overview on what elements of a cloud are manipulated to achieve cloud elasticity. No clue have been found to prove the opposite at the time of writing.

As the proposals are on reacting to variating usage, simulators need a way to express this fluctuating workload. We worked on elastic tasks that model tasks that are triggered regularly and with a usage that fluctuates over time.

Figure 1: Cloud architecture
http://cloud-simulation-frameworks.wikispaces.asu.edu/



III. State of the art

- How workloads are generally modelized
- What simulations for evaluation have been done
- What others evaluations do

Based on the classification of the survey, a simulator should allow the manipulation of scopes, the evaluation of the different purposes, make possible the elastic actions and allow multiple providers.

At the moment no simulator article talks about dynamic workload. On the other hand in the code of DCsim [2] there was an interactive task and in the code of CloudSim [3] there was an host with dynamic workload. There are some tools to generate artificial workload like CITATION NEEDED and they generally follow the following steps. They have a thread that acts like clients/users and it makes request over time and simulate times of thinking of the users.

IV. Contribution

The work was done on SimGrid [4]. The code is available here: https://github.com/sbihel/internship_simgrid. It was written as a plug-in on top of the S4U interface which is intended to be the core API. Elastic tasks are objects and are the only things the user has to manipulate.

An elastic task can repeat a certain task that we will call microtask. The user provides a rate of triggering per second and the flops required and then over time multiple identical microtasks will be created and executed. An elastic task can have multiple hosts to split the workload and there will be a cycling shifting between hosts when creating microtasks, keeping one host for one microtask.

An output function can be provided to an elastic task and this function will be executed after each microtask that has ended. This has multiple usages. It allows the description of workflows of tasks. As microtasks only generate computing workload, output functions can be used to have different types of workloads like network usage, disk access (which can be simulated only by seeing it as a particular computing resource at the moment), and basically anything possible with SimGrid.

It is also useful to study the behavior of a system dealing with real workload. For that an elastic task can be given a file of timestamps and it will trigger/generate a microtask for each time stamp.

For detailed platforms description it is a core feature of SimGrid which allows to have multiple providers, topologies, hosts (VMs for us), bandwidths...

V. Evaluation

The contribution has been evaluated on the predefined criteria. We first did an experiment for raw performances. Then we used real traces from WorldCup 98 data access logs [5] which are often used. After that we evaluated the expressiveness and functionalities. All experiments have been executed on a MacBook Pro with an Intel Core i5 and 8GB of RAM.

A. Raw performances

Figure 2 shows the CPU time (user + system time) while Figure 3 shows the maximum memory used. The platform was upgraded two times, at 1,600 from 2,000 hosts to 100,000, and then at 100,000 from 100,000 to 200,000.

Figure 2: Raw performances CPU time

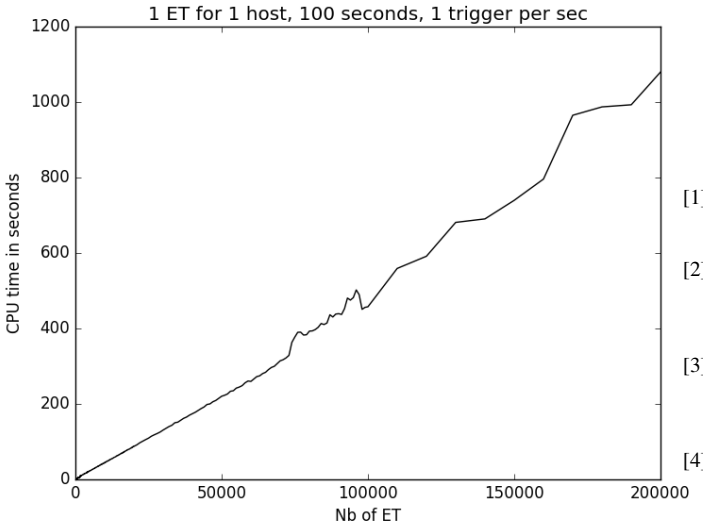
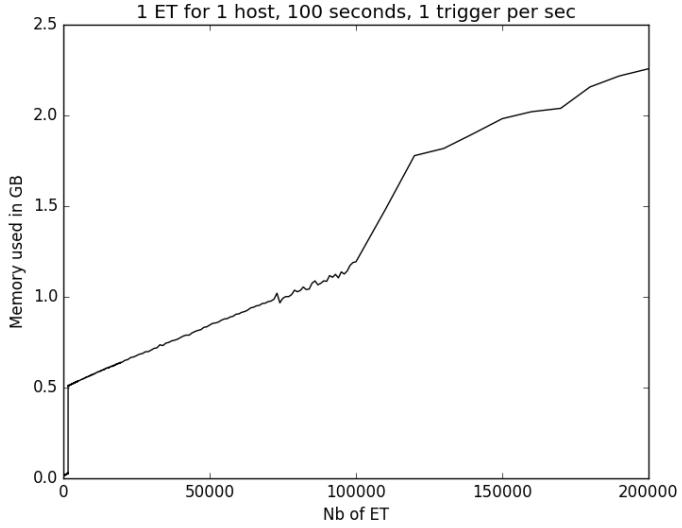


Figure 3: Raw performances Max Memory



B. Real traces

For that we used real traces from WorldCup 98 data access logs [5] and after translating the requests log as a timestamps file we tested with...

C. Functionalities

VI. Future work

VII. Conclusion

References

[1] A. Naskos, A. Gounaris, and S. Sioutas, *Cloud Elasticity: A Survey*. Cham: Springer International Publishing, 2016, pp. 151–167. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-29919-8_12

[2] M. Tighe, G. Keller, J. Shamy, M. Bauer, and H. Lutfiyya, "Towards an improved data centre simulation with dcsim," in *Proceedings of the 9th International Conference on Network and Service Management (CNSM 2013)*. IEEE, 2013, pp. 364–372.

[3] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya, "Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and Experience*, vol. 41, no. 1, pp. 23–50, 2011.

[4] H. Casanova, A. Giersch, A. Legrand, M. Quinson, and F. Suter, "Versatile, scalable, and accurate simulation of distributed applications and platforms," *Journal of Parallel and Distributed Computing*, vol. 74, no. 10, pp. 2899–2917, Jun. 2014. [Online]. Available: http://hal.inria.fr/hal-01017319

[5] "World cup 98 data access logs," http://ita.ee.lbl.gov/html/contrib/WorldCup.html, accessed: 2016-07-13.