Universidade do Minho
Escola de Engenharia

Semana da Escola de Engenharia
October 24 - 27, 2011

Effective Self-Provisioning of Hardware Infrastructure by Assessing Application Scalability

Nuno A. Carvalho and José Pereira
Department of Informatics
E-mail: nuno@di.uminho.pt

KEYWORDS
Scalability, Dependability, Distributed Systems.

ABSTRACT

Assigning virtual machines to physical hosts and allocating specific resources to virtual machines (e.g. processor cores) is a key issue for a cloud computing provider offering Infrastructure-as-a-Service (IaaS). Underprovisioning of resources fails to fulfill the promise of infinite resources. For the provider, it limits the amount of actual usage that can be charged to clients. On the other hand, overprovisioning makes inefficient usage of physical infrastructure. Moreover, provisioning decisions must change as application requirements have to be satisfied dynamically.
In this paper, we show that straightforward usage of reactive adaptation mechanisms based on measurement of CPU core usage often leads to unstable behavior. Trying to avoid such frequent reconfiguration and associated downtime by introducing inertia, results again in sub-optimal resource occupation and failure to adapt timely. We then solve this problem by introducing an adaptation mechanism based on estimation of the system's scalability curve and show that it improves both resource occupation and allows faster reaction to changing requirements.

INTRODUCTION

Currently there is a great complexity between the quality of services provided and the cost of its availability, i.e., by applying strict rules to lower costs imposed by the current business models, but aimed to an growing market, many companies are migrating their products and services solutions to the cloud. This change makes resource management much simpler, both in terms of hardware, it ceases to be a concern, and at the financial level, because is only needed to pay the used resources, eliminating the risk of overbooking or not sufficient hardware.

Yet this simplicity in resource management has a hidden side, the complex issues of provisioning have to be solved by someone, as well as other issues implicit in this business model on the side of the cloud provider, such as how to: 1) know the occupation of the nodes, as well as its characterization, to be able to allocate multiple clients in the most economic and efficient configuration as possible, usually two conflicting objectives; 2) measure the utilization of applications of different customers, without knowing the application itself, that is, each application is a black box; 3) predict what will happen in order to avoid service failures, heavily penalized by service level agreements (SLA).

Most cloud providers rely on fixed adaptation policies, which have been continuously manually adjusted but can not embrace the huge diversity of behaviors displayed by the various applications, which often lead to wrong decisions, or at least not as efficient as desirable. This is compounded by the increasing complexity which characterizes distributed applications, e.g., services provided by telecommunications carriers, or the eternal beta web applications where changes are performed at a dizzying pace. This leads to the key factors affecting the performance change from day to day, but especially they are not those expected either by administrators or by the developers themselves. Which are often called upon to carry out a description of the host application, that only aggravate the wrong decisions, in particular when what is at stake is the allocation in the same machine of various components of a single application.

These issues take on even greater importance when directly affect the amount charged to customers, i.e., the amounts charged depends directly on the resources allocated, if its allocation is not efficient. For example, if are allocated more resources than needed, clients will pay for resources that do not have provided increased performance, being so heavily penalized. Furthermore, these improperly allocated resources could be used to

Semana da Escola de Engenharia
October 24 - 27, 2011

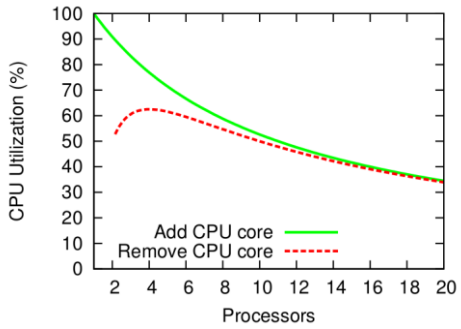

Figure 1: Scalability curve for a given system

accommodate new customers, or turned off, thereby reducing the energy bill.

In this paper we address the problem of achieving both optimal resource occupation and faster reaction to changing requirements related to CPU utilization. The proposed method and tool, through the discovery of the system's scalability curve, is able to decide whether it is really desirable, both in terms of resources and performance, to add or remove CPU cores to or from the system.

SYSTEM SCALABILITY

The main problem of existing adaptation mechanisms is not knowing the scalability limits of the application, that is, they assume that all CPU core additions results in linear increase in performance. The application may be limited by software and not by hardware, the most usual cause nowadays. This omission leads to wrong decisions, or at least not as efficient as desirable.

Our approach [1] fills this assumption, because knowing the scalability curve we know the outcome of the hardware deployment modification before realizing it, thus avoiding unstable or even non-optimal configurations, that without the need to know the application or the need to use benchmarks to define the application behaviour or the safety interval, not needed anymore.

In Figure 1 we can see the scalability curve of a given system, on which is clearly observable the number of CPU cores, from which the configuration is no longer optimal, namely different from 10 cores, setting until each increment has a performance gain greater than or equal to 5%. This threshold on the performance gain of adding a CPU core is a parameter of the cloud provider, being in charge of decide if want more reacting configurations by choosing a small threshold, which will make the adaptation mechanism more sensitive to the workload, or more conservative settings in which will require a major change in workload to trigger adaptations. This decision can also be delegated to the costumer, since it directly influences the quantity of allocated resources, i.e., the lower the threshold the greater the maximum number of allocated resources, having direct influence on the amount charged to the customer.

CONCLUSIONS

In this paper, we propose an adaptation mechanism based on the estimation of the system's scalability curve, and present a set of tools that implement the proposed approach. Our approach provides a effective way to assess and improve resource allocation.
The results show the effectiveness of our approach, cause unlike conventional solutions, it ponders his decisions with the system's scalability curve, achieving more efficient configurations, even allowing for optimal allocation according to different goals such as performance, cost or power consumption.

REFERENCES

[1] N. A. Carvalho and J. Pereira. Measuring software systems scalability for proactive data center management. In R. Meersman, T. Dillon, and P. Herrero, editors, On the Move to Meaningful Internet Systems, OTM 2010, volume 6427 of Lecture Notes in Computer Science, pages 829–842. Springer Berlin / Heidelberg, 2010. 10.1007/978-3-642-16949-6 11.

AUTHORS' BIOGRAPHIES

NUNO A. CARVALHO went to the Universidade do Minho, where he studied Computer Science and Systems Engineering and obtained his degree. He worked as a researcher in the IST FP6 project "GORDA - Open Replication of Databases" and P-SON project "Probabilistically-Structured Overlay Networks". Currently, he is in the MAP-i Doctoral Programme doing his PhD entitled "Self-Managing Service Platform". His e-mail address is: nuno@di.uminho.pt and his web page can be found at http://gsd.di.uminho.pt/members/nac.