Procedia Computer Science
Volume 53, 2015, Pages 232–238
2015 INNS Conference on Big Data

Toward a Framework for Improving the Execution of the Big Data Applications

Wissem Abbes[1,*], Zied Kechaou[1], Adel M. Alimi[1]

[1]*REsearch Groups in Intelligent Machines (REGIM), University of Sfax, National School of Engineers (ENIS), BP 1173, Sfax 3038, Tunisia*
wissem.abbes.tn@ieee.org, zied.kechaou@ieee.org, adel.alimi@ieee.org

Abstract
In this paper, we propose a new framework based on learning techniques to improve the execution of a cloud application, especially the big-data application that requires significant computing capacity. We propose a new metric to detect the available capacity in the cloud client.

Keywords: cloud computing, transparent computing, big-data, learning techniques, resource provisioning

1 Introduction

In the 80's, ubiquitous computing [1] was a dream and with time and the technological progress the dream begins to come true, especially with the advent of transparent computing paradigm [2] in the 2000s. Transparent computing has become a reality and the development has not stopped, another paradigm has emerged that is cloud computing. With the implementation of cloud computing, there is a massive amount of data flowing over the net, the appearance of big data [3] accordingly.

The problem is the response time and quality of service for cloud application, especially the big data application, which requires high computing capacity.

To solve this problem we propose a new framework, based on learning techniques, which is composed of four parts: cloud consumers, resource provisioning, virtual machines and physical machines. The proposed framework utilizes a new metric to detect the available capacity in the cloud client.

The rest of this paper is organized as follows: section 2 defines cloud computing, section 3 defines transparent computing, section 4 presents the relationship between cloud computing and transparent

* Corresponding author. Tel.: +216-74-274-088; fax +216-74-274-088.

Selection and peer-review under responsibility of the Scientific Programme Committee of INNS-BigData2015

doi:10.1016/j.procs.2015.07.299

computing, while section 5 defines big data. Section 6 discusses existing frameworks. We present our proposed framework in section 7, and finally, we present our conclusion and future work in section 8.

2 Cloud computing

Cloud computing is defined by NIST [4] as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. The cloud model is composed of five essential characteristics namely: on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service. It is composed of three models namely: IaaS, PaaS and SaaS. Chee and Curtis [5] define these models as follows:

- IaaS: Infrastructure as a service. It is the service of the lowest level. It is to provide access to a virtualized Information technology infrastructure. The consumer can install an operating system and applications in the virtual machines. The consumer is thus exempt from the purchase of computer equipment. This service is similar to traditional hosting center data processing services, and the trend is for higher-level services, which are more abstract technical details.
- PaaS: Platform as a service. In this type of service, the operating system and infrastructure tools are the responsibility of the supplier. The consumer controls applications and can add his own tools. The situation is analogous to that of the web hosting where the customer rents the operation of servers on which the tools are pre-placed and controlled by the supplier. The difference is that the systems are pooled and offer great elasticity to automatically adapt to demand, whereas in a traditional web hosting offer, adaptation is done following a formal request from the consumer.
- SaaS: Software as a service. In this type of service, applications are made available to consumers. Applications can be manipulated using a web browser, and the consumer does not have to worry about making updates, to add security patches and to ensure service availability.
- Cloud computing has four deployment models namely: private cloud, community cloud, public cloud and hybrid cloud. NIST [4] defines these models as follows:
- Private cloud: The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers. It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.
- Community cloud: The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns. It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.
- Public cloud: The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider.
- Hybrid cloud. The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability.

3 Transparent Computing

Transparent computing is defined by [6] as a computing paradigm in which services are shared. It is considered as a form of persuasive computing.

This paradigm focuses on separating the storage and execution of applications and software, which include operating systems. The data and programs are often executed in servers designed exclusively for clients or the execution of services. Thus, transparent computing helps users share services without interfering with the underlying hardware and compatibility.

Chen and Zheng in [7] define transparent computing as a special kind of cloud computing that provides software storage as a service, or to be more specific, a special kind of IaaS that treats permanent storage as a kind of infrastructure. Transparent computing logically splits the software stack from the underlying hardware platform, and separates the computing unit from the permanent storage for the purpose of making the same software run on different hardware and different software run on the same hardware.

The authors in [2] and [8] define transparent computing as a special kind of cloud computing where storage is treated as a service. The major characteristic of transparent computing involves two separations: the separation of software stack and hardware platform, and the separation of computing and storage. The purpose of transparent computing consists of two runs: to run the same software on different hardware platforms, and to run different software on the same hardware platform. Just like cloud computing, transparent computing also follows the client-server architecture. It is the implementation of pervasive or ubiquitous computing. The execution of computer instruction and data is temporally and spatially separated from their storage.

4 The relationship between cloud computing and transparent computing

In this section, we present a comparison between cloud computing and transparent through the common features and differences. We conclude with a comparative table through three parameters; computing, storage and display.

4.1 Common features

The main common features are as follows:

- Separation of computing and storage.
- Client-server architecture.
- Implementation of SaaS.
- Dynamic management of software resource.
- Transparent software maintenance.
- Less management effort required for a group of clients.

4.2 Differences

The main differences between cloud computing and transparent computing are as follows:

- In cloud computing, the data flow is from server to client: computing and storage is done at the server and displayed locally.
- In transparent computing, data go from client to server: data are generated by local computing and then put in remote storage.

- Transparent computing only utilizes the remote storage, but in cloud computing, everything is finished at the remote server except for the display.
- Transparent computing utilizes the local hardware resources more than the cloud computing [9].

4.3 Cloud computing vs transparent computing

Table 1 summarizes the key differences between cloud computing and transparent computing. [10]

	Cloud computing	Transparent computing
Computing at	Remote	Local
Storage at	Remote	Remote
Display at	Local	Local

Table 1: The relationship between cloud computing and transparent computing

With cloud computing, when the computing is the remote and despite the large computing capabilities found in cloud servers, the high demand of executing an application causes the inability to return a rapid result.

With transparent computing, when the computing is the local, the weak capacity of a terminal e.g. smartphone, tablet or even an old pc; causes the inability to run an application.

5 Big data

According to [11], big data is an all-encompassing term for any collection of data sets, so large and complex that it becomes difficult to process using traditional data processing applications.

The challenges include analysis, capture, curation, search, sharing, storage, transfer, visualization, and privacy violations. The trend to larger data sets is due to the additional information derivable from the analysis of a single large set of related data, as compared to separate smaller sets with the same total amount of data, allowing correlations to be found to “spot business trends, prevent diseases, combat crime and so on”.

In [12] the authors consider that big data is about the growing challenge that organizations face as they deal with large and fast-growing sources of data or information that also present a complex range of analysis and used problems. These can include:

- Having a computing infrastructure that can ingest, validate, and analyze high volumes (size and/or rate) of data.
- Assessing mixed data (structured and unstructured) from multiple sources.
- Dealing with unpredictable content with no apparent schema or structure.
- Enabling real-time or near-real-time collection, analysis, and answers.

Therefore, we conclude, for our work, that the big data applications, analysis or research, are greedy in RAM and CPU.

6 Review of existing frameworks

Many researchers have worked to perform resource provisioning. The application domain considered includes multi-tier web applications, workflow applications or high-performance scientific computing applications. Each application domain has its own quality of service goals, which may either be oriented towards users or service providers.

Time series or machine learning-based approaches are strictly proactive in nature. These techniques involve analysis, training and prediction of metrics based on the past data values [13].

In [14], Huang et al. apply the auto-regression based approach to derive the performance model using resource level parameters and application. The effect of application level parameters including the connection count and the login rate was studied on power consumption and CPU usage to perform load balancing and resource provisioning.

In [15], Bankole and Ajila evaluate the performance of three machines learning techniques, namely, Support Vector Machine (SVM) [16], Neural Networks (NN) [17] and Linear Regression (LR) [18] for the TPC-W benchmark [19] web application. Considering the response time and throughput as the service-level agreement metric their prediction model provides a robust scaling decision. In [20], Sadeka et al. propose a prediction framework that exploits neural networks and linear regression along with a sliding window-based technique for on demand resource allocation in the cloud. For training purposes, they utilized the data generated by executing the TPC-W benchmark in the Amazon EC2 cloud [21].

7 Proposed framework

In our proposed framework, as illustrated in Figure 1, we propose the following parts for improving the execution of an application in the cloud.

1) Physical machines: it is a collection of servers that are connected by a physical network such as a LAN.

2) Virtual machnies: as the name suggests, each virtual machine, called instance, functions like a virtual private server. One of the advantages of virtual machines is the ability to abstract the characteristics of the physical machine used (hardware and software including operating system), allowing a strong software portability and management of legacy systems which are sometimes designed for old machines or software environments and not available.

3) Resource provisioning: in this part, the service request monitor retrieves the needs of the requested service. The available capacity monitor detects the resources available to the cloud customer. Applying the learning techniques such as the LR, SVM or NN the framework predicts the resources to be used in accordance with the service-level agreement.

4) Cloud consumers: A cloud client consists of computer hardware and/or software that relies on cloud computing for application delivery, or that is specifically designed for delivery of cloud services and that, in either case, is essentially useless without it.

When a user requests the execution of a cloud application, the monitor of available capacity begins with the detection of resources available (CPU, RAM) in the user (cloud consumer). According to service request monitor, the requested application will be run in two parts, a part in the client and the other in the cloud.

The use of learning techniques in resource provisioning allows to estimate the resources available for future use of a customer.

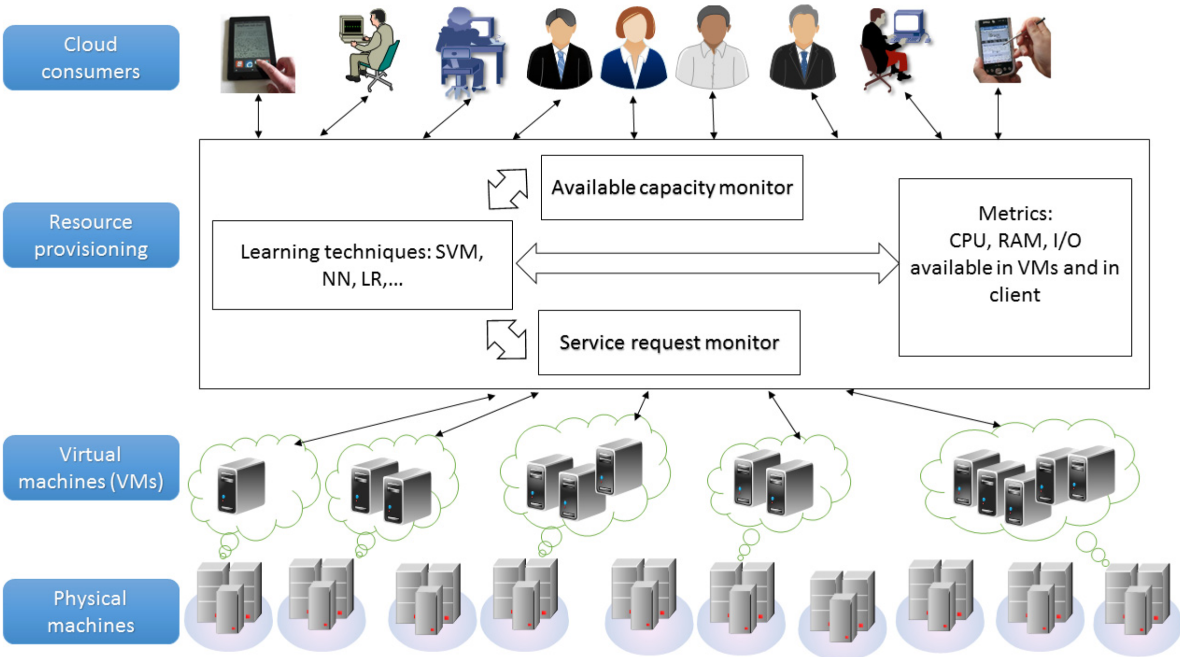

Figure 1: High-level architectural framework

8 Conclusion and future work

In this paper, we defined the concepts of cloud computing and transparent cloud. Then we presented a comparison between these two concepts through the common and difference points. Next, we defined the big data concept.

After that, we have proposed a framework for improving the execution of an application in cloud environment such as the big data application that requires high computing capacity. The framework is based on learning techniques. We proposed a new metric which can detect the available resources, to operate, in the cloud consumer.

As a future work, we plan implementing the framework with various learning techniques. In addition, we will evaluate it on a public cloud.

Acknowledgements

The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

References

[1] Mark Weiser, "Hot Topics: Ubiquitous Computing", IEEE Computer, Oct. 1993.

[2] Y. Zhang, “Transparent computing: concept, architecture and examples”, China Journal of Electronics, Vol. 32, No. 12A, 2004, pp.169–173.

[3] K. Bakshi, “Considerations for big data: Architecture and approach”, in: 2012 IEEE Aerospace Conference, 2012, pp. 1-7.

[4] National Institute of Standards and Technology (NIST).

[5] B. J.S. Chee, F. Jr. Curtis, “Cloud computing: technologies and strategies of the ubiquitous data center,” CRC Press, 2010.

[6] Techopedia website. [online]. Available: http://www.techopedia.com/definition/14610/transparent-computing, 30-09-2014.

[7] K. Chen and W. Zheng, “Transparent computing: ahead of the cloud computing”, China Computer World, Vol. 17, 2008.

[8] Y. Zhou and Y. Zhang, “Transparent computing: concept, architecture and implementation”, Cengage Learning Press, Singapore, 2010.

[9] Y. Zhang and Y. Zhou, “Stream Software Architecture”, communications of China Computer Association, 2010, pp.6-11.

[10] W. Ming, “Analysis and a case study of transparent computing implementation with UEFI”, Int. Journal of Cloud Computing, Vol. 1, No. 4, 2012, pp.312–328.

[11] D. Boyd and K. Crawford, “Critical questions for big data”, Information, Communication & Society, 15:5, 2012, pp. 662-679, doi:10.1080/1369118X.2012.678878

[12] L. R. Villars, C. W. Olofson and M. Eastwood, “Big data: what it is and why you should care”, IDC, Jun. 2011.

[13] P. D. Kaur, I. Chana, “A resource elasticity framework for QoS-aware execution of cloud applications”, Future Generation Computer Systems 37, 2014, pp. 14-25.

[14] J. Huang, C. Li, J. Yu, “Resource prediction based on double exponential smoothing in cloud computing”, IEEE 2nd International Conference on Consumer Electronics, Communications and Networks, CECNet, 2012, pp. 2056–2060.

[15] A. A. Bankole, S. A. Ajila, “Predicting cloud resource provisioning using machine learning techniques”, 26th Annual IEEE Canadian Conference on Electrical and Computer Engineering(CCECE), 2013, pp. 1 - 4.

[16] B. Schölkopf, A. J. Smola, “Learning With Kernels: Support Vector Machines, Regularization, Optimization and Beyond”, 2002, MIT Press.

[17] D. E Rumelhart., and J. L McClelland., “Parallel Distributed Processing: Explorations in the Microstructure of Cognition”, MIT Press, Cambridge, 1986, pp. 567-568.

[18] N.R Draper and H. Smith, “Applied Regression Analysis”, John Wiley, 3rd ed., 1998.

[19] TPC, TPC-W Benchmark, Transaction Processing Performance Council (TPC), San Francisco, CA, USA, 2003.

[20] I. Sadeka, K. Jacky, L. Kevin, L. Anna, “Empirical prediction models for adaptive resource provisioning in the cloud”, Future Generation Computer Systems 28, 2012, pp. 155-165.

[21] “Amazon elastic compute cloud (amazon ec2)”. [Online]. Available: http://aws.amazon.com/ec2/faqs, 2014.