International Journal of Modern Trends in Engineering and Research
www.ijmter.com
e-ISSN No.:2349-9745, Date: 2-4 July, 2015

# Resource Mapping Analysis for Many Task Computing on Cloud

**Brij Raj Singh, Prof. Rajesh Bharati**

[1] *Department of Computer Engineering, Padmashree Dr. D.Y. Patil Institute of Engineering & Technology, University of Pune, India, brijraj2011@gmail.com*
[2] *Professor, Department of Computer Engineering, Padmashree Dr. D.Y. Patil Institute of Engineering & Technology, University of Pune, India , rdbharati@gmail.com*

**Abstract**: Cloud computing has emerged as a very important commercial infrastructure that promises to reduce the need for maintaining costly computing facilities by organizations and institutes. Through the use of virtualization and time sharing of resources, clouds serve with a single set of physical resources as a large user base with altogether different needs. Thus, the clouds have the promise to provide to their owners the benefits of an economy of calibration and, at the same time, become a substitute for scientists to clusters, grids, and parallel production conditions. However, the present commercial clouds have been built to support web and small database workloads, which are very different from common scientific computing workloads. Furthermore, the use of virtualization and resource time sharing may introduce significant performance penalties for the demanding scientific computing workloads. In this paper, we analyze the performance of cloud computing services for scientific computing workloads. In this paper, we analyze different resource allocation techniques for cloud computing that are now present. We also propose improvements in certain resource allocation techniques**.**

**Keywords:** Cloud Computing, Scientific Computing, distributed applications, performance evaluation, metrics/measurement, performance measures.

## 1. INTRODUCTION

Scientific computing requires an ever-increasing number of resources to deliver results for ever-growing problem sizes in a reasonable time frame. In the last decade, while the largest research projects were able to afford (access to) expensive supercomputers, many projects were forced to opt for cheaper resources such as commodity clusters and grids. Cloud computing proposes an alternative in which resources are no longer hosted by the researchers' computational facilities, but are leased from big data centers only when needed. Despite the existence of several cloud computing offerings by vendors such as Amazon and GoGrid, the potential of clouds for scientific computing remains largely unexplored. To address this issue, in this paper we present a performance analysis of cloud computing services for many-task scientific computing.

The cloud computing paradigm holds great promise for the performance-hungry scientific computing community: Clouds can be a cheap alternative to supercomputers and specialized clusters, a much more reliable platform than grids, and a much more scalable platform than the largest of commodity clusters. Clouds also promise to "scale by credit card," that is, to scale up instantly and temporarily within the limitations imposed only by the available financial resources, as opposed to the physical limitations of adding nodes to clusters or even supercomputers and to the administrative burden of over provisioning resources. Moreover, clouds promise good support for bags-of-tasks (BoTs), which currently constitute the dominant grid application type [3]. However, clouds also raise important challenges in many aspects of scientific computing, including performance, which is the focus of this work.

There are three main differences between scientific computing workloads and the initial target workload of clouds: in required system size, in performance demand, and in the job execution model. Size wise, top scientific computing facilities comprise very large systems, with the top ten entries in the Top500 Supercomputers List together totaling about one million cores, while cloud computing services were designed to replace the small-to-medium size enterprise data centers. Performance wise, scientific workloads often require High-Performance Computing (HPC) or High-Throughput Computing (HTC) capabilities. Recently, the scientific computing community has started to focus on Many-Task Computing (MTC), that is, on high performance execution of loosely coupled applications comprising many (possibly interrelated) tasks. With MTC, a paradigm at the intersection of HPC and HTC, it is possible to demand systems to operate at high utilizations, similar to those of current production grids (over 80 percent) and Parallel Production Infrastructures (PPIs) (over 60 percent), and much higher than those of the systems that clouds originally intended to replace (servers with 10-20 percent utilization). The job execution model of scientific computing platforms is based on the exclusive, space-shared usage of resources. In contrast, most clouds time-share resources and use virtualization to abstract away from the actual hardware, thus increasing the concurrency of users but potentially lowering the attainable performance.

## 2. LITERATURE SURVEY

Improving the performance of cloud computing services for scientific computing is the fundamental task in several cloud providers. We have selected for this work four IaaS clouds. The reason for this selection is threefold. First, not all the clouds on the market are still accepting clients; FlexiScale puts new customers on a waiting list for over two weeks due to system overload. Second, not all the clouds on the market are large enough to accommodate requests for even 16 or 32 coallocated resources. Third, our selection already covers a wide range of quantitative and qualitative cloud characteristics

Srirama S.N, Ivanistsev V, Jakovits P, Willmore C[1] designed a tool that helps scientists to migrate their applications to the cloud. The idea is to migrate the complete software environment, in which the scientists have set up their experiments, directly to the cloud. The developed desktop-to-cloud-migration (D2CM) toolsupports transformation and migration of virtual machine images, deployment description and life-cycle management for applications to be hosted on Amazon's Elastic Cloud Computing (EC2)or compatible infrastructure such as Eucalyptus. They also presented an electrochemical case study which extensively used the tool in drawing domain specific results. From the analysis, it was observed that D2CM tool not only helps in the migration process and simplifying the work of the scientist, but also helps in optimizing the calculations, compute clusters and thus the costs for conducting scientific computing experiments on the cloud. Guang Lin, Han Binh, Yin Jian, I Gorton [2] explored cloud computing for large-scale data intensive scientific applications. Cloud computing is attractive because it provides hardware and software resources on-demand, which relieves the burden of acquiring and maintaining a huge amount of resources that may be used only once by a scientific application. However, unlike typical commercial applications that often just requires a moderate amount of ordinary resources, large-scale scientific applications often need to process enormous amount of data in the terabyte or even petabyte range and require special high performance hardware with low latency connections to complete computation in a reasonable amount of time. To address these challenges, we build an infrastructure that can dynamically select high performance computing hardware across institutions and dynamically adapt the computation to the selected resources to achieve high performance. We have also demonstrated the effectiveness of our infrastructure by building a system biology application and an uncertainty quantification application for carbon sequestration, which can efficiently utilize data and computation resources across several institutions. simon Ostermann, Alexandria Iosup, Nezih Yigitbasi, Radu Prodan, Thomas Fahringer, Dick Epema [3] presented an evaluation of the usefulness of the current cloud computing services for scientific computing. They analyze the performance of the Amazon EC2 platform using micro-benchmarks and kernels. While clouds are still changing, their results indicate that the current cloud services need an order of magnitude in performance improvement to be useful to the scientific community. P. Jakovits, SN. Srirama, I. Kromonov [4] presented a design for a new distributed computing framework, Stratus, which fully supports scientific computing algorithms and takes advantage of the characteristics that have made cloud such a convenient and popular source for computing resources. They also described the motivation for creating a brand new solution, outlines its architecture and design, and gives an overview on how algorithms can be adapted to this framework. Iosup, O.O. Sonmez, S. Anoep, and D.H.J. Epema [5] defined Bags-of-tasks that have been identified as the scientists' tool of choice, but there is no workload model that explicitly includes bags-of-tasks. Here they make a realistic and systematic investigation of the performance of bags-of-tasks scheduling solutions in large-scale distributed computing systems. They first propose a taxonomy of scheduling policies that focuses on information availability and accuracy, and they propose three new classes bagof-tasks scheduling policies; for each new class we also propose a simple task scheduling policy. Then, they introduce a realistic workload model that focuses on bags-of-tasks, and we validate this model using seven long-term workload traces taken from large-scale distributed computing systems of various size and application. Finally, they explore the large design space of bag-of-task scheduling in large-scale distributed computing systems along five axes: the scheduling policy, the input workload, the information policy, the scheduling algorithm, and the resource management architecture. S. Ostermann, R. Prodan, T. Fahringer [6] investigated the usability of compute Clouds to extend a Grid workflow middleware and show on a real implementation that this can speed up executions of scientific workflows. From its start using supercomputers, scientific computing constantly evolved to the next levels such as cluster computing, meta-computing, or computational Grids. Today, Cloud Computing is emerging as the paradigm for the next generation of large-scale scientific computing, eliminating the need of hosting expensive computing hardware. Scientists still have their Grid environments in place and can benefit from extending them by leased Cloud resources whenever needed. This paradigm shift opens new problems that need to be analyzed, such as integration of this new resource class into existing environments, applications on the resources and security. The virtualization overheads for deployment and starting of a virtual machine image are new factors which will need to be considered when choosing scheduling mechanisms.

## 3. BASIC SYSTEM ARCHITECTURE

We first describe the main characteristics of the common scientific computing workloads, based on previous work on analyzing and modeling of workload traces taken from PPIs[6] and grids[5], [13].
Then, we introduce the cloud computing services that can be used for scientific computing, and select four commercial clouds whose performance we will evaluate empirically. job structure and source - PPI workloads are dominated by parallel jobs [6], while grid workloads are dominated by small bags-of-tasks [3] and sometimes by small workflows [14], [15] comprising mostly sequential tasks. Source wise, it is common for PPI grid workloads to be dominated by a small number of users. We consider users that submit many tasks, often grouped into the same submission as BoTs, as proto-MTC users, in that they will be most likely to migrate to systems that provide good

performance for MTC workload execution .bottleneck resources. Overall, scientific computing workloads are highly heterogeneous, and can have either one of CPU, I/O, memory, and network as the bottleneck resource, Job parallelism. A large majority of the parallel jobs found in published PPI [16] and grid [13] traces have up to 28 processors [5], [6]. Moreover, the average scientific cluster size was found to be around 32 nodes [17] and to be stable over the past five years [18].
With the emergence of cloud computing as a paradigm in which scientific computing can done exclusively on resources leased only when needed from big data centers, Relative Strategy Performance: Resource Bulk Allocation (S2) versus Resource Acquisition and Release per Job (S1) Only performance differences above 5 percent are shown. e-scientists are faced with a new platform option. However, the initial target workloads of clouds do not match the characteristics of MTC-based scientific computing workloads. Thus, in this paper we seek to answer the research question Is the performance of clouds sufficient for MTC-based scientific computing? To this end, we first investigate the presence of an MTC component in existing computing workloads, and find that this presence is significant both in number of jobs and in resources consumed. Then, we perform an empirical performance evaluation of four public computing clouds, including Amazon EC2, one of the largest commercial clouds currently in production. Our main finding here is that the compute performance of the tested clouds is low. Last, we compare the performance and cost of clouds with those of scientific computing alternatives such as grids and parallel production infrastructures. We find that, while current cloud computing services are insufficient for scientific computing at large, they may still be a good solution for the scientists who need resources instantly and temporarily.

## CONCLUSION

In this way we surveyed performance of different cloud computing services for scientific computing. This paper contains an abstract view of various cloud platforms proposed in recent past year for scientific computing. The investigation demonstrated that better performance of cloud is possible for just about any additional techniques and some changes. Our contribution towards this work will surely be helpful for further improving the performance of clouds for scientific computing. This paper quantify the presence in real scientific computing workloads of Many-Task Computing (MTC) users, that is, of users who employ loosely coupled applications comprising many tasks to achieve their scientific goals.

## REFERENCES

[1] S.N. Srirama, V. Ivanistsev, P. Jakovits, C. Willmore, "Direct migration of scientific computing experiments to the cloud" High Performance Computing and Simulation (HPCS), 2013 International Conference, 5 July 2013, pp. 27-34.
[2] Guang Lin, Han Binh, Yin Jian, I Gorton, "Exploring Cloud Computing for Large-Scale Scientific Applications", Services (SERVICES), 2013 IEEE Ninth World Congress, July 3 2013 pp. 37-43
[3] Simon Ostermann, Alexandria Iosup, Nezih Yigitbasi, Radu Prodan, Thomas Fahringer, Dick Epema, "A Performance Analysis of EC2 Cloud Computing Services for Scientific Computing" Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering Volume 34, 2010, pp. 115-131
[4] P. Jakovits, S.N. Srirama, I. Kromonov, "Stratus: A Distributed Computing Framework for Scientific Simulations on the Cloud" High Performance Computing and Communication & 2012 IEEE 9th International Conference, 27 June 2012.
[5] A. Iosup, O.O. Sonmez, S. Anoep, and D.H.J. Epema, "The Performance of Bags-of-Tasks in Large-Scale Distributed Systems," Proc. ACM Int'l Symp. High Performance Distributed Computing (HPDC), pp. 97-108, 2008.
[6] S. Ostermann, R. Prodan, T. Fahringer, "Extending Grids with cloud resource management for scientific computing" Grid Computing, 2009 10th IEEE/ACM International Conference, p. 42-49, 2009.

 

[7] I. Raicu, Z. Zhang, M. Wilde, I.T. Foster, P.H. Beckman, K. Iskra, and B. Clifford, "Toward Loosely Coupled Programming on Petascale Systems," Proc.ACM Conf. Supercomputing (SC), p. 22, 2008.
[8] A. Iosup, C. Dumitrescu, D.H.J. Epema, H. Li, and L. Wolters, "How Are Real Grids Used? The Analysis of Four Grid Traces and Its Implications," Proc. IEEE Seventh Int'l Conf. Grid Computing, pp. 262-269, 2006.
[9] U. Lublin, D.G. Feitelson, "Workload on Parallel Supercomputers: Modeling Characteristics of Rigid Jobs," J. Parallel and Distributed Computing, vol. 63, no. 11, pp. 1105-1122, 2003.
[10] L. Youseff, R. Wolski, B.C. Gorda, and C. Krintz, "Paravirtualization for HPC Systems," Proc. ISPA Workshops, pp. 474-486, 2006.
[11] E. Deelman, G. Singh, M. Livny, J.B. Berriman, and J. Good, "The Cost of Doing Science on the Cloud: The Montage Example," Proc. IEEE/ACM Supercomputing (SC), p. 50, 2008.
[12] M.R. Palankar, A. Iamnitchi, M. Ripeanu, and S. Garfinkel, "Amazon S3 for Science Grids: A Viable Solution?" Proc. DADC '08: ACM Int'l Workshop Data-Aware Distributed Computing, pp. 55-64, 2008.
[13] E. Walker, "Benchmarking Amazon EC2 for HP Scientific Computing," Login, vol. 33, no. 5, pp. 18-23, Nov. 2008.
[14] L. Wang, J. Zhan, W. Shi, Y. Liang, L. Yuan, "In Cloud, Do Mtc or Htc Service Providers Benefit from the Economies of Scale?" Proc. Second Workshop Many- Task Computing on Grids and Supercomputers (SC-MTAGS), 2009.
[15] A. Iosup, H. Li, M. Jan, S. Anoep, C. Dumitrescu, L. Wolters, and D. Epema, "The Grid Workloads Archive," Future Generation Computer Systems, vol. 24, no. 7, pp. 672-686, 2008.
[16] D. Thain, J. Bent, A.C. Arpaci-Dusseau, R.H. Arpaci- Dusseau, and M. Livny, "Pipeline and Batch Sharing in Grid Workloads," Proc. IEEE 12th Int'l Symp. High Performance Distributed Computing(HPDC), pp. 152-161, 2003.
[17] S. Ostermann, A. Iosup, R. Prodan, T. Fahringer, and D.H.J. Epema, "On the Characteristics of Grid Workflows," Proc. Workshop Integrated Research in Grid Computing (CGIW), pp. 431-442, 2008.
[18] A. Iosup, O.O. Sonmez, and D.H.J. Epema, "DGSim: Comparing Grid Resource Management Architectures through Trace-Based Simulation," Proc. 14th Int'l Euro-Par Conf. Parallel Processing, pp. 13-25, 2008..

## AUTHOR PROFILE

**Brij Raj Singh**
Research Scholar Dr. D. Y. Patil Institute of Engineering & Technology, Pune, University of Pune, Maharashtra, India. He has received his Bachelor's Degree in Computer Science from Army Institute if Technology, Pune, Pune University, Maharashtra. Currently He is pursuing his Master's Degree in Computer Engineering from Dr. D. Y. Patil Institute of Engineering & Technology, Pune, University of Pune.

**Prof. Rajesh Bharati**
Presently working as a Professor in Department of Computer Engineering, Dr. D. Y. Patil Institute of Engineering and Technology, Pimpri, Pune, Maharstra, India.

## ACKNOWLEDGMENT

I would like to take this opportunity to acknowledge the contribution of certain people without which it would not have been possible to complete this paper work. I am thankful to the Principal Dr. R. K. Jain, Guide, Head, Coordinators, Colleagues of the Department of Computer Engineering, Dr. D. Y. Patil Institute of Engineering and Technology, Pimpri, Pune, Maharashtra, India, for their support, encouragement and suggestions. I would like to express my special appreciation and thanks to my guide Professor Prof. Rajesh Bharati, you have been a tremendous mentor for me.