Convergence of Distributed Clouds, Grids and their Management – CDCGM2012

WETICE2012 – Toulouse, June 25 – 27, 2012

Track Chair's Report

Dr. Rao Mikkilineni
IEEE Member
Kawa Objects Inc.,
Los Altos, California, USA
rao@kawaobjects.com

Dr. Giovanni Morana
DIEEI
University of Catania
Catania, Italy
giovanni.morana@dieei.unict.it

***Abstract*— We restate the objective of the first workshop on Collaboration and Cloud Computing" in WETICE 2009; "to analyze current trends in Cloud Computing and identify long-term research themes and facilitate collaboration in future research in the field that will ultimately enable global advancements in the field that are not dictated or driven by the prototypical short term profit driven motives of a particular corporate entity." We are glad to report that the discussions started in 2009 have directly resulted in an alternative approach to self-managing distributed computing systems totally different from current industry trend showing a way to eliminate the complexity of virtual machines and Hypervisors. If this approach is proven to be theoretically sound (as a paper in this conference is investigating) and extend its usefulness (demonstrated through their feasibility in the form of two proofs of concepts in the last conference) to mission critical environments, the DIME network architecture may yet prove to be an important contribution to computer science. In addition to the two papers related to the DIME computing model, there are nine other long papers and two short papers selected out of twenty two submissions discussing various aspects of clouds, grids and their management.**

Keywords-component; Cloud Computing; grid computing; Distributed Intelligent Managed Element Networks; Distributed Services Management; Services Virtualization; Parallel Computing; Many-core Servers

I. INTRODUCTION

Clouds, Grids and their management are the center of attention of both the academicians and the industry as the recent number of conferences and publications demonstrate. Since the inception of current track in 2009, along with a rapid growth and acceptance of cloud computing, three disturbing trends have developed in cloud technology landscape driven by the short term profit motives of an overly competitive industry:

1. **Proliferation of Hypervisors and ad hoc Orchestrators:** the virtualization of computing has mushroomed into multiple Hypervisors, some directly virtualizing at the bare-metal and others incorporated into various operating systems. In order to grab the market share each operating system is combined with a particular brand of virtualization to support other operating systems as guest operating systems. This has created a new complexity with in the physical server by necessitating a virtual network inside the physical server. The economics of multi-tenancy inside the physical server are now being off-set by the complexity of coordination and management of virtual servers.
2. **Proliferation of ad hoc Storage and Database Strategies:** In order to deal with structured and un-structured data that is growing at a rate of five trillion bits per second, various storage strategies are being invented adding to the complexity of storage management. Lack of formal models to deal with storage distribution is forcing ad hoc approaches to deal with replication, concurrency, and consistency. New databases that are schema-free with easy replication and eventual consistency are being developed. While these approaches solve specific problems in a niche area, the resulting complexity in enterprise is only increased.
3. **Proliferation of ad hoc Management Software:** In the name of open-source software that is supposed to provide protection against vendor lock-in and foster competition, recent efforts to develop new cloud management platforms such as CloudStack and OpenStack have resulted in crowd-sourced software explosion to compete with already existing large base of existing platforms from VMWare, Citrix, Microsoft, IBM and other large players. The result is software that ends up as shelf-ware more often than not. In addition, the need for expertise and professional services to manage the open-source complexity has also exploded making IT more expensive.

1524-4547/12 $26.00 © 2012 IEEE
DOI 10.1109/WETICE.2012.79

As a result, while the complexity of the management infrastructure has exploded, there still is no clear solution for providing end to end distributed transaction availability, performance and security that spans across multiple virtual servers, data centers and geographies. End-to-end distributed service transaction visibility and control today, is only possible if all the infrastructure providers provide the information necessary for management that spans across multiple vendor products, heterogeneous operating environments and a plethora of management systems. No wonder, 70% to 80% of today's Information Technology budget is being consumed to maintain itself and is not available to create new services that either increase revenue or improve quality of products and services.

With the expected increase of number of computing devices to tens of millions in the data centers of the future (Google already uses about a million servers), current approach of designing systems where errors will become as conspicuous as possible, and intervention and correction follow immediately is untenable as von Neumann pointed out in his 1948 Hixon Lectures [1].

It is important to take a step back and look at the root cause for this complexity. Resource management based data center operations miss an important feature of services/applications management which is that all services are not created equal. They have different latency and throughput requirements. They have different business priorities and different workload characteristics and fluctuations. Figure 1 shows a classification of different services based on their throughput and latency requirements.

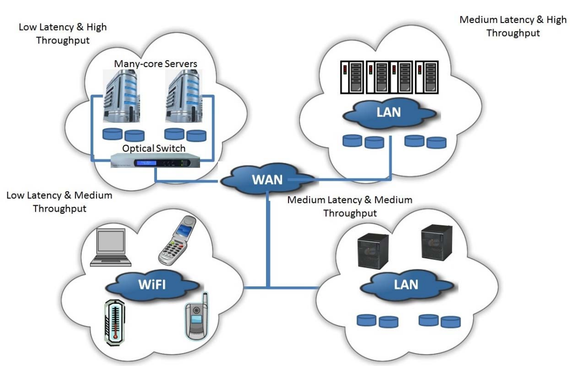

Figure 1: A distributed service transaction spans across computing devices, geographies and spans of control. Each service has its own characteristic that demands various resources with different latency and throughput constraints.

Distributed transactions transcend the abilities of current serial stored program control implementation of the Turing machine which is at the heart of the atomic computing element in current computing infrastructure. The communication is not an integral part of this atomic computing unit in the stored program control implementation of the Turing machine. The distributed transactions require interaction, which integrates computing, and communication to provide the ability to specify and execute highly temporal and hierarchical events.

According to Goldin and Wegner, Interactive computation is inherently concurrent, where the computation of interacting agents or processes proceeds in parallel. Hoare, Milner and other founders of concurrency theory have long realized that Turing Machines (TM) do not model all of computation (van Leeuwen and Wiedermann, 2000), (Wegner and Goldin, 2003), (Wegner and Eberbach, 2003), (Wegner and Eberbach, 2004). "However, when their theory of concurrent systems was first developed in the late '70s, it was premature to openly challenge TMs as a complete model of computation. Their theory positions interaction as orthogonal to computation, rather than a part of it. By separating interaction from computation, the question whether the models for CCS and the Pi-calculus went beyond Turing Machines and algorithms was avoided.

The resulting divide between the theory of computation and concurrency theory runs very deep. The theory of computation views computation as a closed-box transformation of inputs to outputs, completely captured by Turing Machines. By contrast, concurrency theory focuses on the communication aspect of computing systems, which is not captured by Turing Machines – referring both to the communication between computing components in a system, and the communication between the computing system and its environment. As a result of this division of labor, there has been little in common between these fields and their communities of researchers. According to Papadimitriou (Papadimitriou, 1995), such a disconnect within the theory community is a sign of a crisis and a need for a Kuhnian paradigm shift in our discipline."

With the advent of many-core processors with parallel hardware threads and high bandwidth communications among the Turing machines, current operating systems and distributed system management strategies with their server-centric origins are ready for a reexamination (Mikkilineni 2011). Autonomic computing discipline is attempting to address some of these deficiencies. Alan Ganek (Ganek, 2007) points out that autonomic computing is not to be confused with automation. "Autonomic computing focuses on enabling systems to adjust and adapt automatically based on business policies. It addresses the process of how IT infrastructures are designed, managed and maintained." Different architectures and many implementations have contributed to the progress of the autonomic computing discipline (Denko, Yang and Zhang, 2009). Object, component, service and agent orientation paradigms have emerged as major paradigms for implementing autonomous distributed systems. Nonetheless, it is argued (Braubach & Pokahr, 2011) that "none of these paradigms is able to adequately describe all kinds of distributed systems and inherent conceptual limitations exist." While each paradigm addresses some aspects of distributed computing, none of them are capable of supporting concurrency, distribution, and non-functional aspects such as availability, reliability, performance, scalability and end-to-end transaction security. These authors propose an extension by combining agent and service component architecture to create an active component which provides a hierarchical composition scheme to implement managed workflows. However, all these ad hoc approaches to distributed computing depend on their implementations on an SPC implementation of the Turing machine which is constrained by its serial implementation of computations and the prohibition of precise specification, self-reflection, and execution together of a workflow by the SPC implementation of the Turing machine by itself

Kuhnian paradigm shift or not, the first two papers (Eberbach, Mikkilineni and Morana), (Goyal and Mikkilineni) address the DIME network architecture and the new computing model discussed in last year's conference. In Section II, we summarize these two papers. In Section III, we summarize the other nine papers dealing with various management issues with cloud and Grid computing. In Section IV, we summarize three short papers that potentially could lead to further interesting results in the future.

II. New Computing Models and a Path to Self-Managing Distributed Computing Systems

The plenary talk and the paper by Eugene Eberbach review the various computing models available today and their inadequacy to model autonomic distributed computing systems. The author has contributed many papers with Peter Wegner and others calling for a paradigm shift by extending the current Turing machine based approach to computer science. His attempt to examine the process algebras and in particular π- and $-calculi in detail to describe the DIME computing features of replication, repair, reconfiguration and recombination is just the beginning to evaluate new approaches.

The second paper by Goyal argues that the current set of Business Process Management Systems do not support the dynamicity of relationships and processes – or process mobility – creating new channels, reassigning channels, in particular, to processes whose existence becomes known only at run-time and may differ by instance. This dynamic nature of the relationships and processes is founded on π-calculus mobility. This paper presents a new approach to implement π-calculus mobility using DIME network architecture; it is shown how process mobility is an inherent capability of DIMEs. This paper also proposes the use of DIME FCAPS capabilities to support business services management, including fault tolerance, performance and security.

Hopefully, these theoretical investigations will stimulate other investigations to examine the usefulness and the soundness of the DIME network architecture in decreasing the complexity and improving the architectural resiliency of distributed transactions.

III. Cloud And Grid Computing Management Issues

While two papers mentioned above are concerned with next generation distributed computing, six papers are devoted to address current cloud and grid optimization issues on the practical side.

The first three are devoted to addressing the performance aspects:

1. Performance prediction of parallel programs over Cloud infrastructures is an important issue. The authors (Cuomo, Rak and Villano) discuss the practical validity of the distribution of the concurrent simulation workload to computing resources hosted in a cloud. They demonstrate how, given a scalable trace (i.e., a trace valid for different number of processors), the required simulations can be straight forwardly executed in a single step.
2. The paper "Decentralized Resource Finding in Cloud/Grid Computing Environments: a Performance Evaluation" by Messina, Pappalardo, and Santoro, deals with the topic of resource finding in a very large Grid or Cloud Computing system. The results provided in the paper show that ``informed" heuristics, i.e. those which take into account the proposed network organization and the metrics introduced, perform quite better than random approaches, and such a difference is much more evident in the case of highly loaded system where the availability of resources is scarce.
3. Calvagna, Pappalardo and Tramontana describe an MPI-based parallel approach for scalable combinatorial interaction testing of large requirements spaces.

Two papers are devoted to service oriented architecture issues in distributed clouds:

1. Following a paper presented in WETICE 2011, Omezzine, Yangui, Bellamine, and Tata discuss Mobile service micro-containers for Cloud environments. Their addition of service mobility at the micro-container level suggests improvements over current Virtual Machine migration. Another interesting opportunity arises with the introduction of DIME network architecture to investigate encapsulation of service micro-containers in DIMEs.
2. With the wide acceptance of cloud computing (both private and public clouds), some enterprises are looking at a combination (hybrid clouds) to optimize their resource utilization. The paper by Charrada, Tebourski, Tata, and Moalla address the problem of placing service-based applications in hybrid Clouds.

A very interesting application of High Performance computing and Grid-MPI infrastructure to chart a three-dimensional map of our Galaxy, the Milky Way is described by Bandieramonte, Becciani, Vecchiato, Lattanzi, and Bucciarelli. They report that a hybrid solution which uses a modified PC-LSQR, a conjugate gradient-based algorithm, with the aid of some parallelization techniques and of an ad-hoc compression algorithm of the sparse System Matrix, the performance has reached the theoretical maximum of the speed-up curve (doubling the Pus halves the time).

A paper by Di Sano, Morana and Zito describes an innovative approach for providing a file system-like management to transient and consistent set of files shared among cooperating applications in clouds.

Giunta, Messina, Pappalardo, and Tramontana present an Aspect-Oriented architecture to improve the Quality of Service of a Web Server.

The paper by Punceva, Rodero, Parashar, and Petri describes an economical-inspired solution for incentivizing resource sharing in Social Clouds.

IV. Short Papers

WETICE prides itself in encouraging graduate students and researchers bringing new ideas that have not yet been accepted by the main stream or authors who would like to present yet-incomplete research to get feedback from experts to present their ideas in the form of short papers. It is hoped that the following discussions will result in some new research papers in the future.

In this track, two short papers are presented:

1. Personal Knowledge Advantage Machine: Pervasive Methods for Personal Semantic Information Retrieval by Desjardins, Moparthi, Vangela, Reddy, and Reddy
2. Galaxy workflow integration on Grid GARUDA by Karuna and Mangala

V. Conclusion

This year's conference continues the tradition of WETICE to follow its objective set at its inception; "what sets WETICE apart from larger conferences is that the conference tracks are kept small enough to promote fruitful discussions on the latest technology developments, directions, problems, and requirements." This year, we selected eleven papers and three short papers from a pool of twenty two submissions. The new computing model discussion is very timely, on the eve of Turing centenary celebration, to address some fundamental issues in distributed computing to show a new path to self-managing systems and hopefully will reduce the current complexity that is exploding in our data centers.

We hope that some of the ideas discussed here will be carried further and some new areas of research will blossom to be presented at the next WETICE track on the convergence of distributed clouds, grids and their management.

Acknowledgment

The Chair persons thank many of the reviewers and the WETICE organizing committee for their support.

References

[1] Braubach, L., Pokahr, A., (2011), "Addressing Challenges of Distributed Systems Using Active Components" in Intelligent Distributed Computing V, New York, NY: Springer

[2] Denko, M. K., Yang, L. T., Zhang, Y., (2009). "Autonomic Computing and Networking", New York, NY, Springer.

[3] Ganek, A., (2007). "Overview of Autonomic Computing: Origins, Evolution, Direction" in Autonomic Computing: Concepts, Infrastructure, Applications. Edited by Parashar, M., and Hariri, S., Boca Raton, FL:CRC Press, pp.3

[4] Mikkilineni, R., (2011) "Designing a New Class of Distributed Systems," New York, Springer, ISBN: 1461419239

[5] Neumann, J. v., (1987) General and Logical Theory of Automata. edited and compiled by William Aspray and Arthur Burks, MIT Press, p408

[6] van Leeuwen, J., Wiedermann, J., (2000) "The Turing machine paradigm in contemporary computing" In B. Enquist and W. Schmidt, Eds., Mathematics Unlimited—2001 and Beyond.LNCS, Springer-Verlag, 2000.

[7] Wegner, P., Eberbach, E., (2004). New Models of Computation. The Computer Journal, vol 47, No. 1, 4-9.

[8] Wegner, P., Goldin, D., (2003). Computation beyond Turing Machines: Seeking appropriate methods to model computing and human thought. Communications of the ACM, Vol. 46, No. 4, pp. 100

[9] Goldin D., Wegner P., (2008) "The Interactive Nature of Computing: Refuting the Strong Church–Turing Thesis", Minds & Machines 18:17–38, DOI 10.1007/s11023-007-9083-1

[10] Eberbach E., Wegner P., (2003) "Beyond Turing Machines," The Bulletin of the European Association for Theoretical Computer Science (EATCS Bulletin), 81, Oct., pp. 279-304.