2015 IEEE 24th International Conference on Enabling Technologies: Infrastructures for Collaborative Enterprises

Track Chair's Report of Convergence of Distributed Clouds, Grids and their Management (CDCGM-2015)

Back to the Future: Disruptive Innovation and the Convergence of Cognition, Computing and Communications

Fabrizio Messina
Dipartimento di Matematica e Informatica,
Università di Catania
V.le Andrea Doria 6, 95125 Catania, Italy
messina@dmi.unict.it

Rao Mikkilineni
IEEE Member,
C^3DNA Inc.,
Santa Clara, USA
rao@c3dna.com

Giovanni Morana
Head of R&D and Engineering,
C^3DNA Inc., Catania, Italy
giovanni.morana@c3dna.com

***Abstract*—This track started with the first Cloud Computing session in WETICE2009 with the observation that the cloud computing evolution depends on research efforts from the infrastructure providers creating next generation hardware that is service friendly, service developers that embed business service intelligence in the network to create distributed business workflow execution, assure service delivery on a massive scale with global interoperability while dealing with non-functional requirements such as security, availability, performance, compliance, cost and fluctuations both in resources and workloads. It was pointed out that the architecture and evolution of the cloud is increasing datacenter complexity by piling up new layers of management over the many layers that already exist. Current session proves the epigram by Jean-Baptiste (1849) "plus ça change, plus c'est la même chose." Literally "The more it changes, the more it's the same thing". In this conference, one paper presents the result of the discussions started in these sessions in 2009 that led to a policy based dynamic workflow orchestration independent of the infrastructure orchestration which eliminates the need for moving Virtual Machine images and interfaces to myriad infrastructure management systems at runtime. The architecture is derived from the well-understood and time-tested distributed systems such as cellular organisms, human organizational structures and telecommunication networks. In addition, eight full papers and three short papers continue to make progress on current state of the art. Based on the papers presented in these sessions over the last six years, we boldly predict that we are on the verge of a synthesis of the thesis of current state of the art and the anti-thesis of increasing complexity to address scaling and fluctuations in distributed systems.**

***Keywords*—Cloud Computing; grid computing; Distributed Intelligent Managed Element Networks; Distributed Services Management; Services Virtualization; Parallel Computing;**

I. Introduction

According to Holbrook [1], [2], "*Specifically, creativity in all areas seems to follow a sort of dialectic in which some structure (a thesis or configuration) gives way to a departure (an antithesis or deviation) that is followed, in turn, by a reconciliation (a synthesis or integration that becomes the basis for further development of the dialectic). In the case of jazz, the structure would include the melodic contour of a piece, its harmonic pattern, or its meter [. . .]. The departure would consist of melodic variations, harmonic substitutions, or rhythmic liberties [. . .]. The reconciliation depends on the way that the musical departures or violations of expectations are integrated into an emergent structure that resolves deviation into a new regularity, chaos into a new order, and surprise into a new pattern as the performance progresses*" Current IT in this Jazz metaphor, evolved from a thesis of client server computing and currently is experiencing an anti-thesis due to demands of scale and fluctuations both in workloads and available pools of finite resources. It is ripe for a synthesis that would blend the old and the new with a harmonious melody to create a new generation of highly scalable, distributed, secure services with desired availability, cost, compliance and performance characteristics to meet the changing business priorities, highly fluctuating workloads, and latency constraints.

The IT Thesis. Current generation server, networking, storage elements and related software systems for management have evolved from server-centric and bandwidth limited network architectures to today's cloud computing platforms with virtual servers and broadband networks [3], [4]. During last six decades, many layers of computing abstractions have been introduced to map the execution of complex computational workflows to a sequence of 1s and 0s that eventually get stored in the memory and operated upon by the CPU to achieve the desired result. These include process definition languages, programming languages, file systems, databases, operating systems etc. While this has helped in automating many business processes, the exponential growth in services in the consumer market also has introduced severe strains on current IT infrastructure [5]. In order to meet the need to rapidly respond to manage the distributed computing resources demanded by changing workloads, business priorities and latency constraints, new layers of resource management are added with the introduction of Hypervisors, virtual machines (VM) and their management [6]–[9]. While these layers have made the application or service management more agile, they have introduced a new layer of issues related to their own management. For example, new layers of VM-level clustering, application intrusion detection and performance management,

978-1-4673-7692-1/15 $31.00 © 2015 IEEE
DOI 10.1109/WETICE.2015.40

are being introduced in addition to already existing clusters, intrusion detection and performance management systems at the infrastructure, operating systems and distributed resource management layers [10].

IT Anti-Thesis. The origin of complexity is easy to understand. While attempting to solve the issue of multi-tenancy and agility, the introduction of Virtualization [11]–[13] gives rise to another complexity of virtual image management, movement and sprawl control [14]. In order to address VM mobility issue, recent efforts to introduce application level mobility using other container constructs such as Dockers, LXC, LXD, CoreOS etc., along with associated Infrastructure as a Service (IaaS) and Platform as a Service (PaaS), all of which go to great lengths, to provide High Availability (HA) of the Infrastructure platform [15]. These ad-hoc approaches to automate management have proliferated the software required, increased the learning curve and made the operation and maintenance even more complex. While all platforms demonstrate drag and drop software with pretty displays that allow developers to easily create new services, there is no guarantee that if something goes wrong, one will be able to debug and find out where the root cause is. Or there is no assurance that when multiple services and applications are deployed on same platform, the feature interactions and shared resource management provided by a plethora of management systems designed independently will cooperate to provide the required reliability, availability, performance and security at the service level. More importantly, when the services cross server, data-center and geographical boundaries, there is no visibility and control of end to end service connections and their fault, configuration, accounting, performance and security (FCAPS) management. Obviously, the platform vendors are very eager to provide professional services and additional software to resolve the issues but without end to end service connection visibility and control that spans across multiple modules, systems, geographies and management systems, troubleshooting expenses often outweigh the realized benefits. What we need probably is not more "code" but an intelligent architecture that results in a synthesis of computing services and their management and a decoupling of end to end service connection and service component management from underlying resource (server, network and storage) management.

IT Synthesis. A business process is defined both by functional requirements that dictate the business domain functions and logic as well as non-functional requirements that define operational constraints related to service availability, reliability, performance, security and cost dictated by business priorities, workload fluctuations and resource latency constraints. A non-functional requirement specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. The plan for implementing functional requirements is detailed in the system design. The plan for implementing non-functional requirements is detailed in the system architecture. While much progress has been made in the system design and development, the architecture of distributed systems falls short to address the non-functional requirements for two reasons:

• Current distributed systems architecture from its server-centric and low-bandwidth origins has created layers of resource management-centric ad-hoc software to address various uncertainties that arise in a distributed environment. Lack of support for concurrency, synchronization, parallelism and mobility of applications dictated by the current serial Von-Neumann stored program control has given rise to ad-hoc software layers that monitor and manage distributed resources. While this approach may have been adequate when distributed resources are owned by a single provider and controlled by a framework that provides architectural support for implementing non-functional requirements, the proliferation of commodity distributed resource clouds offered by different service providers with different management infrastructures adds scaling and complexity issues. Current OpenStack and AWS API discussions are a clear example that forces a choice of one or the other or increased complexity to use both.

• The resource-centric view of IT currently demotes application and service management to a second-class citizenship where the QoS of application/service [16] is monitored and managed by myriad resource management systems overlaid with multiple correlation and analysis layers used to manipulate the distributed resources [17]–[19] to adjust the CPU, memory, bandwidth, latency, storage IOPs, throughput and capacity which are all what are required to keep the application/service to meet its quality of service demands as a function of time. Obviously, this approach cannot scale unless single set of standards evolve or a single vendor lock-in occurs.

There are three factors influencing a major change in how enterprises are seeking to reduce complexity, shorten time to market and time to fix with end-to-end service visibility and control to manage their service offering's availability, performance, security, compliance and cost:

1) Commodity infrastructure in the form of distributed, highly scalable computing clusters available on demand from multiple service providers at competitive prices with desired Service Levels in terms of node CPU, memory, network bandwidth, latency and storage capacity, IOPs and throughput,
2) New meta-container technology that decouples end to end service transaction quality of service (QoS) that spans from the end user device to the backend database with little needed regard for infrastructure management that involves virtual machine image motion between source and target environments to provide desired service levels, and
3) A service composition and policy based workflow orchestration using managed application components with a service control channel that is an overlay over the data path that connects the read/write communications between application components [20]

The DIME network architecture [21] which had its origins in WETICE has proven to be a vehicle to provide a synthesis of existing IT with the new computing, programming and management models that allow a smooth transition to address distributed service transactions with required resiliency, scaling and efficiency without disrupting current IT operations.

II. Summary of the Session

There are eight long papers and three short papers dealing with current datacenter thesis and antithesis while 1 paper deals with the synthesis bringing a fresh new approach to distributed clouds, grids and their management.

Cognitive Application Area Networks: A New Paradigm for Distributed Computing and Intelligent Service Orchestration by Rao Mikkilineni, Giovanni Morana and Daniele Zito, describes a model and a method to capture the non-functional requirements to manage the resources and implement policy based adjustment of the hardware and software to assure the availability, performance, security, compliance and cost optimizations based on business priorities, latency constraints and workload fluctuations. The paper claims implementing self-repair, auto-scaling and live migration of distributed systems with or without virtualization.

An evolutionary approach for Cloud learning agents in multi-cloud distributed contexts by Fabrizio Messina, Giuseppe Pappalardo, Domenico Rosaci, Giuseppe M.L. Sarné, Lidia Fotia and Antonello Comi presents an evolutionary approach based on agent cloning, i.e. a mechanism of agent reproduction allowing providers to substitute an "unsatisfactory" agent acting in a "cloud context" with a clone of an existing agent having a suitable knowledge and a good reputation in the multi-cloud context. By this approach, cloud agents performances can be improved because they are substituted with agent clones that have shown a better behavior.

A Reputation-based approach to improve QoS in Cloud Service Composition by Fabrizio Messina, Giuseppe Pappalardo, Domenico Rosaci, Giuseppe M.L. Sarnè, Lidia Fotia and Antonello Comi proposes a reputation-based model capable to support the composition of complex services by considering costs and measures of QoS which are collected by measuring systems, and reputation collected from the customers.

REST-based SLA Management for Cloud Applications by Alessandra De Benedictis, Massimiliano Rak, Mauro Turtur and Umberto Villano describes the design of services for the management of cloud-oriented SLAs that hinge on the use of a REST based API. Such services can be easily integrated into existing cloud applications, platforms and infrastructures, in order to support SLA-based cloud services delivery.

A Resource Allocation Model Driven Through Quality of Experience by André D'amato, Mario Dantas and Douglas Macedo presents a proposal of a new model to allocate resources for grids called MAROQ that uses context information. Experimental results show that the use of context information improved the average execution time of a task in 7.46%.

Cloud Service Matchmaking using Constraint Programming by Begüm Ilke Zilci, Mathias Slawik and Axel Küpper develop a concept for a service matcher which contributes to existing approaches by addressing these issues using constraint solvers. This allows service requesters with limited technical knowledge to be able to compare services based on their QoS requirements in cloud service marketplaces.

A Self-Optimized Storage for Distributed Data as a Service by Klaithem Al Nuaimi, Nader Mohamed, Mariam Alnuaimi and Jameela Al-Jaroodi present the design for 3 a self-optimized storage for distributed data as a service.

Clustering EU's Countries According to I. Th. Mazi's Systemic Geopolitical Theory Using K-means and MPI by Ilias Savvas, Alekos Stogiannos and Ioannis Mazis confirms the results of clustering EU's countries according to I. Th. Mazi's systemic geopolitical theory using K-means and MPI.

Load Index Characterization: The Scientific Challenge, and a Survey Approach by Guilherme Maciel Ferreira and Mario Dantas addresses the scientific challenge of load index characterization to reduce current complexity in scheduling of program executions in cloud computing. Their work uses a formal method to search, select and synthesize the most prominent publications in order to present a differentiated survey of the most relevant load indices available in the literature. They propose a set of categories to classify those publications, indicating the state of art of this field.

The three short papers focus on identifying the issues with current cloud computing architectures in assuring quality of service (availability, performance, security, compliance and cost) and attempt to investigate alternatives to reduce complexity.

III. Conclusions

This year, one paper has opened alternatives to current cloud computing complexity reduction and eight long and three short papers that address various aspects of cloud, grid and distributed computing issues. The continued discussion of the new computing model is very timely to address some fundamental issues in distributed computing to show a new path to self-managing systems and hopefully will stimulate more research. We conclude this paper with this quotation from Mark Burgin [22]. The "gap between the hardware and the software of a concrete computer and even greater gap between pure functioning of the computer and its utilization by a user, demands description of many other operations that lie beyond the scope of a computer program, but might be represented by a technology of computer functioning and utilization". As Mark points out, knowing the intent, monitoring and managing the application, applying best-practice knowledge to adjust to fluctuations, and changing the circumstance must be part of the service management knowledge independent of distributed infrastructure management systems for providing true scalability, distribution and resiliency. This avoids vendor lock-in or infrastructure, architecture or API lock-in [23].

Acknowledgment

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

References

[1] M. B. Holbrook, "Adventures in complexity: An essay on dynamic open complex adaptive systems, butterfly effects, self-organizing order, co-evolution, the ecological perspective, fitness landscapes, market spaces, emergent beauty at the edge of chaos, and all that jazz," *Academy of Marketing Science Review*, vol. 6, no. 1, pp. 1–184, 2003.

[2] ——, *Playing the Changes on the Jazz Metaphor*. Now Publishers Inc, 2008.

[3] R. Buyya and K. Sukumar, "Platforms for building and deploying applications for cloud computing," *arXiv preprint arXiv:1104.4379*, 2011.

[4] F. Messina, G. Pappalardo, and C. Santoro, "Integrating cloud services in behaviour programming for autonomous robots," in *Algorithms and Architectures for Parallel Processing*. Springer International Publishing, 2013, pp. 295–302.

[5] R. Mikkilineni and V. Sarathy, "Cloud computing and the lessons from the past," in *Enabling Technologies: Infrastructures for Collaborative Enterprises, 2009. WETICE'09. 18th IEEE International Workshops on*. IEEE, 2009, pp. 57–62.

[6] F. Messina, G. Pappalardo, D. Rosaci, and G. M. Sarné, "A trust-based, multi-agent architecture supporting inter-cloud vm migration in iaas federations," in *Internet and Distributed Computing Systems*. Springer International Publishing, 2014, pp. 74–83.

[7] F. Messina, G. Pappalardo, D. Rosaci, C. Santoro, and G. M. Sarné, "A trust model for competitive cloud federations," *Complex, Intelligent, and Software Intensive Systems (CISIS)*, pp. 469–474, 2014.

[8] F. Messina, G. Pappalardo, C. Santoro, D. Rosaci, and G. M. Sarne, "An agent based negotiation protocol for cloud service level agreements," in *WETICE Conference (WETICE), 2014 IEEE 23rd International*. IEEE, 2014, pp. 161–166.

[9] F. Messina, G. Pappalardo, D. Rosaci, and G. M. Sarne, "An agent based architecture for vm software tracking in cloud federations," in *2014 Eighth International Conference on Complex, Intelligent and Software Intensive Systems (CISIS)*. IEEE, 2014, pp. 463–468.

[10] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of internet services and applications*, vol. 1, no. 1, pp. 7–18, 2010.

[11] M. Rosenblum, "Vmware's virtual platform™," in *Proceedings of hot chips*, vol. 1999, 1999, pp. 185–196.

[12] A. Kivity, Y. Kamay, D. Laor, U. Lublin, and A. Liguori, "kvm: the linux virtual machine monitor," in *Proceedings of the Linux Symposium*, vol. 1, 2007, pp. 225–230.

[13] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the art of virtualization," *ACM SIGOPS Operating Systems Review*, vol. 37, no. 5, pp. 164–177, 2003.

[14] V. Sarathy, P. Narayan, and R. Mikkilineni, "Next generation cloud computing architecture: Enabling real-time dynamism for shared distributed physical infrastructure," in *Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE), 2010 19th IEEE International Workshop on*. IEEE, 2010, pp. 48–53.

[15] N. Kratzke, "A lightweight virtualization cluster reference architecture derived from open source paas platforms."

[16] R. Giunta, F. Messina, G. Pappalardo, and E. Tramontana, "Kaqudai: a dependable web infrastructure made out of existing components," in *Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2013 IEEE 22nd International Workshop on*. IEEE, 2013, pp. 146–151.

[17] F. Messina, G. Pappalardo, and C. Santoro, "Decentralised resource finding in cloud/grid computing environments: A performance evaluation," in *Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2012 IEEE 21st International Workshop on*. IEEE, 2012, pp. 143–148.

[18] ——, "Exploiting the small-world effect for resource finding in p2p grids/clouds," in *20th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*. IEEE, 2011, pp. 122–127.

[19] ——, "Decentralised resource finding and allocation in cloud federations," in *Intelligent Networking and Collaborative Systems (INCoS), 2014 International Conference on*. IEEE, 2014, pp. 26–33.

[20] F. Messina, G. Pappalardo, and E. Tramontana, "Design and evaluation of a high-level grid communication infrastructure," *Concurrency and Computation: Practice and Experience*, vol. 19, no. 9, pp. 1299–1316, 2007.

[21] R. Mikkilineni, "Dime network architecture: Future research directions and conclusion," in *Designing a New Class of Distributed Systems*, ser. SpringerBriefs in Electrical and Computer Engineering. Springer New York, 2011, pp. 59–65.

[22] M. Burgin, *Super-recursive algorithms*. Springer Science & Business Media, 2006.

[23] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, and J. Molina, "Controlling data in the cloud: outsourcing computation without outsourcing control," in *Proceedings of the 2009 ACM workshop on Cloud computing security*. ACM, 2009, pp. 85–90.