
Editorial

Hamid Mcheick*

University of Quebec At Chicoutimi (UQAC),
555 Boul. De l'Université, Chicoutimi, G7H-2B1, Canada
E-mail: hamid_mcheick@uqac.ca
*Corresponding author

Mehdi Adda

University of Quebec At Rimouski (UQAR),
300 allée des Ursulines, Rimouski (QC), G5L 3A1, Canada
E-mail: adda@ieee.org
E-mail: mehdi_adda@uqar.ca

Biographical notes: Hamid Mcheick is currently an Associate Professor in the Computer Science Department at the University of Quebec At Chicoutimi (UQAC), Canada. He holds a Master and PhD in Software Engineering and Distributed System from Montreal University, Canada. He is interested in software development and architecture for enterprise applications as well as in separation of concerns (component, services, aspect, etc.). He has given many talks in congress and conferences as a keynote or invited speaker. He is the Editor-in-Chief and Guest Editor of several international journals. His research is supported by many research grants that he has received from the Canadian Government, University of Montreal, Centre de Recherche informatique de Montreal (CRIM), University of UQAM and University of Quebec at Chicoutimi, Canada.

Mehdi Adda is a Regular Professor of Computer Science at the University of Quebec At Rimouski (UQAR, Canada) since 2010. From 2008 to 2010, he was an Invited Professor at the same university. His principal research interests lie in the fields of software, mobile and web engineering, data mining and knowledge discovery, aspect-oriented programming and distributed computing, web personalisation and recommendation. He obtained in 2008 two PhDs in Computer Science; one from the University of Montreal (Montreal, Qc, Canada) and one from the University of Lille (Lille, France). He has two MSs in Computer Science from Joseph Fourier University in 2002 (Grenoble, France) and the University du Havre (Le Havre, France) in 2003; and an engineering degree in Computer Science from the University of Sciences and Technology Houari Boumediène (Algiers, Algeria) in 2001.

Cloud computing (CC) has attracted the attention not only of the scientific community, but also that of large technology companies, application developers and technology consumers around the world with its central promise to offer computing on demand and intuitively. CC presents a new, convenient and flexible way of thinking, using, and managing computing resources and services. The following quote from Armbrust et al. (2009) summarises well the concept of CC: “the long dreamed vision of computing as a utility is finally emerging. The elasticity of a utility matches the need of businesses

providing services directly to customers over the Internet, as workloads can grow (and shrink) far faster than 20 years ago”.

CC is a popular new concept of using virtualisation to outsource IT infrastructure of server farms. The emergence of this new concept as a viable solution for deploying virtual machines instead of physical machines was favoured by relatively low prices of computer hardware and increase network capacity. Based on the concept of cloud computing infrastructure, different services may be provided (Mell and Grance, 2009):

- 1 Software as a service (SAAS): The CC structure provides an application service to customers through composable services that run on this infrastructure. Clients are end users of these services and do not care about the installation, configuration and maintenance of software and platforms on which they run. As examples of SAAS services, we cite Google Docs, Zoho, Salesforce.com, etc.
- 2 Platform as a service (PAAS): Development platforms, usually based on high-level programming languages such as python, ruby, etc., are offered as services to developers. Of such services, we cite Amazon Elastic Beanstalk, Google App Engine, Heroku, etc.
- 3 Infrastructure as a service (IAAS): Unlike in PAAS, in IAAS, the user needs to instal, configure and deploy the runtime environment of his/her applications. Amazon EC2 and Microsoft Azure are among existing public IAAS services.

A more general definition of CC is given in Armbrust et al. (2009); where CC is defined as a computing style that enables organisations access to a shared and elastic set of resources (servers, networks, storage, applications, services, etc.). It aims to ease resource management and provisioning, and reduce service provider interaction.

The main characteristics and preoccupations of CC are: scalability and elasticity, virtualisation, geographic distribution and homogeneity, service orientation, security and privacy, ubiquitous network access, etc. Computing style in CC must meet the computing needs, such as abstraction, dynamism and resources sharing.

- 1 Abstraction: Focus on core business instead of worrying about the infrastructure, its security and updates. Those are left to CC provider.
- 2 Dynamism: Dynamically add new computing resources as business grows.
- 3 Resources: CC is built from the ground up or resources sharing. It is one of the promises of CC: optimise resource utilisation.

Being just in its infancy, CC is not without limitations and issues that are still open, it offers new opportunities with their batches of challenges (Jeffery and Neidecker-Lutz, 2010). These challenges are discussed in this special issue.

In this special issue of *International Journal of Communication Networks and Distributed Systems (IJCND)*, the aim is to publish research contributions that significantly advance the state-of-the-art research in development through CC and service oriented computing to improve software quality of information systems.

Acknowledgements

The Guest Editors would like to thank all the members of the programme committee for their helpful comments, suggestions and evaluations of the articles submitted to *IJCNDIS* 2012, special issue – cloud computing.

References

- Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I. and Zaharia, M. (2009) *Above the Clouds: A Berkeley View of Cloud Computing*, Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley, Tech. Rep. UCB/EECS, Vol. 28, February.
- Jeffery, K. and Neidecker-Lutz, B. (Eds.) (2010) ‘The future of cloud computing – opportunities for European cloud computing beyond 2010’, January, available at <http://cordis.europa.eu/fp7/ict/ssai/docs/cloud-report-final.pdf> (accessed on 11 June 2012).
- Mell, P. and Grance, T. (2009) *NIST Definition of Cloud Computing*, NIST: National Institute of Standards and Technology, US Department of Commerce, Special Publication 800-145, October.