SOFTWARE – PRACTICE AND EXPERIENCE
Softw. Pract. Exper. 2012; **42**:391–394
Published online 9 November 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/spe.1144

Special section: software architectures and application development environments for Cloud computing

Welcome to the special issue of *Software: Practice and Experience* journal on Cloud computing. This special issue compiles a number of excellent technical contributions that significantly advance the state-of-the-art of software architectures and application development environments for cloud computing.

Cloud computing [1–3] is positioning itself as a promising platform for delivering infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), and software-as-a-service (SaaS) as services. Clouds aim to power the next-generation data centers by architecting them as a network of virtual services (hardware, database, user-interface, application logic) so that users are able to deploy and access applications globally and on demand at competitive costs depending on users' QoS requirements. Cloud infrastructures are exposed through collections of software services at SaaS and PaaS layers designed to support creation and deployment of application services. To this end, developing scalable architectures and application development environments to build, access, manage, deploy, and maintain applications in clouds in a developer-friendly manner has become critical.

Several vendors have emerged in this space including IBM, VMware, Microsoft, Manjrasoft, and Yahoo. This model of computing is quite attractive, especially for small and medium size enterprises, as it allows them to focus on consuming or offering services on top of the Cloud infrastructure. At high level, Cloud computing might not seem radically different from the existing paradigms: World Wide Web, grid computing, service computing, and cluster computing. However, key differentiators of Cloud computing are its technical characteristics such as on-demand resource pooling or rapid elasticity, self-service, almost infinite scalability, end-to-end virtualization support, and robust support of resource usage metering and billing. Additionally, nontechnical differentiators include services that are offered under pay-as-you-go-model, guaranteed SLA, faster time to deployments, lower upfront costs, little or no maintenance overhead, and environment friendliness.

Public IaaS and PaaS vendors including Amazon, Microsoft, Google, and GoGrid offer different types of software programming architectures and interfaces. Next, these are implemented using different programming environments, hence should be accessed through vendor-dependent adapter interfaces. In particular current Cloud programming approaches have the following limitations: (i) requires human familiarity with different types of Cloud resources and typically rely on procedural programming in general purpose or scripting languages; (ii) interaction with Cloud resources is mainly performed through low-level APIs and command line interfaces; (iii) SaaS (application) implementation is dependent on the programming environment supported by the IaaS and Paas vendors; and (iv) lacks flexibility and efficiency of supporting generic applications that can be simultaneously deployed across multiple Cloud vendors infrastructure. Hence, it is clear that developing system architecture and application development environments that can simplify and improve the task of Cloud programming are key to harnessing the capability of clouds. In this special issue, we have featured high quality papers that deal with some of the aforementioned issues. All of the selected papers underwent a rigorous peer-review process and their contributions are briefly discussed below:

Though Cloud computing infrastructure services enable the flexible creation of virtual infrastructures on demand basis, it is only a tiny step of the overall complex process required for provisioning application services. Other steps such as installation, deployment, configuration, monitoring, and management of software components are needed to fully provide services to end-users in the Cloud. To this end, in the paper titled 'Towards an Architecture for Deploying Elastic Services in the Cloud', Kirschnick *et al.* [4] describes a peer-to-peer architecture to automatically deploy services on cloud infrastructures. The architecture uses a component repository to manage the deployment of these software components, enabling elasticity by using the underlying cloud infrastructure provider. The life cycle of these components is described in this paper, as well as the language for defining them. They also describe the open-source proof-of-concept implementation. Some technical information about this implementation together with some statistical results is also provided.

Software distributions within the cloud are subject to security breaches, privacy abuses, and access control violations. Illegal copying, malicious tampering, and other infringements on Cloud hosted services, applications, and data are some of the examples of such security, privacy, and trust threats. To tackle the issues of access control, Yu *et al*. in the paper titled ‘A Novel Watermarking Method for Software Protection in the Cloud’ [5], identify an insider threat to access control, which is not completely eliminated by the usual techniques of encryption, cryptographic hashes, and access-control labels. They address this threat using software watermarking. The authors evaluate the access-control scheme within the context of a collaboration-oriented architecture, as defined by The Jericho Forum.

Protection of users’ privacy is critical and has become one of the most concerned issues as otherwise users may eventually lose the confidence and passion of deploying application services on virtualized Cloud services (e.g., CPU, storage, databases, etc.). Under some special cloud circumstances, some users’ privacy, such as plans or habits, could be induced from their service requests by service providers without permissions from users. In this regard, obfuscation strategy can protect this kind of privacy by injecting ‘noise’ service requests to confuse potential ‘immoral’ service providers. However, the existing noise obfuscation strategies focus on single noise injection, whereas investigation of noise injection architecture has been neglected. Especially, a common service pattern in interclouds environment, the cooperative service process including different service providers, makes the risk of privacy serious and uncontrollable by the spread of users’ privacy. To address this, Zhang *et al*. [6] in the paper titled ‘A Trust-based Noise Injection Strategy for Privacy Protection in Cloud’ present a novel trust-based noise injection strategy for privacy protection in the Cloud. To support the strategy, they describe their noise injection architecture in the Cloud, which specializes in the relations between various service roles in interclouds based on a trust model. The simulation can demonstrate that their noise injection strategy could significantly improve the effectiveness of privacy protection.

Recall that Cloud computing promises a radical shift in the provisioning of computing resources within the enterprise. There are a number of challenges involved with adoption of clouds by enterprise as the decisions on migrating IT services to the cloud should not only simply be driven by cost considerations but should also cater for a range of socio-technical factors into account. To aid the decision-making process, authors in the paper titled ‘The Cloud Adoption Toolkit: Supporting Cloud Adoption Decisions in the Enterprise’ [7] present a Cloud Adoption Toolkit that supports decision making during the adoption of Cloud computing in enterprises. The toolkit provides a framework to support decision makers in identifying their concerns, and matching these concerns to appropriate tools/techniques that can be used to address them. Cost Modeling is the most mature tool in the toolkit, and this paper shows its effectiveness by demonstrating how practitioners can use it to examine the costs of deploying their IT systems on the Cloud. The Cost Modeling tool is evaluated using a case study of an organization that is considering the migration of some of its IT systems to the Cloud. The case study shows that running systems on the Cloud using a traditional ‘always on’ approach can be less cost effective, and the elastic nature of the Cloud has to be used to reduce costs. Therefore, decision makers have to model the variations in resource usage and their systems’ deployment options to obtain accurate cost estimates.

As a cost-effective and time-efficient way to develop new applications and services, service aggregation in cloud computing empowers all service providers and consumers, and creates tremendous opportunities in various industry sectors. However, it also poses various challenges to the privacy of personal information as well as the confidentiality of business and governmental information. The full benefits of service aggregation in Cloud computing would only be enjoyed if the privacy concerns are addressed. In the paper titled ‘Privacy Preserving Protocol for Service Aggregation in Cloud Computing’, Wang *et al*. [8] investigate the privacy issues in service aggregation for the Cloud environment, and propose a privacy preserving protocol that is suitable for this environment.

To demonstrate the security of their system, they construct a security game called IND-P3SAC-CPA and prove the security of protocol accordingly. Their protocol has a distinct property, which is that any service provider only obtains the data under its conspiracy with the cloud. Additionally, the efficiency and various extensions are also discussed.

System virtualization is the enabling technology to manage the increasing number of different applications inside Cloud data centers. The abstractions from the underlying hardware and the provision of multiple virtual machines (VM) on a single physical server have led to a consolidation and more efficient usage of physical servers. The abstraction from the hardware also eases the provision of applications on different data centers, as applied in several Cloud computing environments. In this case, the application need not adapt to the environment of the Cloud computing provider, but can travel around with its own VM image, including its own operating system and libraries. System virtualization and cloud computing could also be very attractive in the context of high-performance computing (HPC). Today, HPC centers have to cope with both, the management of the infrastructure and also the applications. Virtualization technology would enable these centers to focus on the infrastructure, while the users, collaborating inside their virtual organizations (VOs), would be able to provide the software. Nevertheless, there seems to be a contradiction between HPC and cloud computing, as there are very few successful approaches to virtualize HPC centers. In the paper titled 'Virtualized HPC: a contradiction in terms', Birkenheuer *et al.* [9] discuss the underlying reasons, including the management and performance, and presents solutions to overcome the contradiction, including a set of new libraries. The viability of the presented approach is shown based on evaluating a selected parallel, scientific application in a virtualized HPC environment.

One of the key challenges that hold businesses from adopting Cloud computing services is that, by migrating to the Cloud, they move some of their information and services out of their direct control. Their main concern is how well the Cloud providers keep their information (security) and deliver their services (performance). To cope with this challenge, several service level agreement management systems have been proposed. However, monitoring service deployment as a major responsibility of those systems have not been deeply investigated yet. In their paper, Dastjerdi *et al.* [10] show how monitoring services have to be described, deployed (discovered and ranked), and then how they have to be executed to enforce accurate penalties by eliminating service level agreement failure cascading effects on violation detection.

We hope that the readers will find the papers of this special issue to be informative and useful.

REFERENCES

1. Armbrust M, et al. A view of Cloud Computing. *Communications of the ACM Magazine* 2010; **53**(4):50–58.
2. Buyya R, Broberg J, Goscinski A (eds). *Cloud Computing: Principles and Paradigms*. Wiley Press: USA, February 2011. ISBN-13: 978-0470887998.
3. Wang L, Ranjan R, Chen J, Benatallah B. *Cloud Computing: Methodology, Systems, and Applications*. CRC Press, Taylor and Francis Group, 2011. ISBN: 9781439856413.
4. Kirschnick J, Alcaraz Calero JM, Goldsack P, Farrell A, Guijarro J, Loughran S, Edwards N, Wilcock L. Towards an Architecture for Deploying Elastic Services in the Cloud. *Software: Practice and Experience* 2011. DOI: 10.1002/spe.1090.
5. Yu Z, Wang C, Thomborson C, Wang J, Lian S, Vasilakos AV. A Novel Watermarking Method for Software Protection in the Cloud. *Software: Practice and Experience* 2011. DOI: 10.1002/spe.1088.
6. Zhang G, Yang Y, Yuan D, Chen J. A Trust-based Noise Injection Strategy for Privacy Protection in Cloud. *Software: Practice and Experience* 2011. DOI: 10.1002/spe.1052.
7. Khajeh-Hosseini A, Greenwood D, Smith J, Sommerville I. The Cloud Adoption Toolkit: Supporting Cloud Adoption Decisions in the Enterprise. *Software: Practice and Experience* 2011. DOI: 10.1002/spe.1072.
8. Wang P, Mu Y, Susilo W, Yan J. Privacy Preserving Protocol for Service Aggregation in Cloud Computing. *Software: Practice and Experience*. DOI: 10.1002/spe.1129.
9. Birkenheuer G, et al. Virtualized HPC: *a contradiction in terms? Software: Practice and Experience* 2011. DOI: 10.1002/spe.1055.
10. Dastjerdi AV, Tabatabaei S, Buyya R. A Dependency-aware Ontology-based Approach for Deploying Service Level Agreement Monitoring Services in Cloud. *Software: Practice and Experience*. DOI: 10.1002/spe.1104.

RAJIV RANJAN
CSIRO Information and Communication Technologies (ICT) Centre
Information Engineering Laboratory
North Road, Acton, ACT 2601, Australia

RAJKUMAR BUYYA
Cloud Computing and Distributed Systems (CLOUDS) Laboratory
Department of Computer Science and Software Engineering
The University of Melbourne, Australia

BOUALEM BENATALLAH
Service Oriented Computing Research Group
School of Computer Science and Engineering
The University of New South Wales, Australia

DOI: 10.1002/spe