Taming the Cloud: Safety, Certification and Compliance for Software Services

Keynote at the Workshop on Engineering Service-Oriented Applications (WESOA) 2011

Howard Foster and George Spanoudakis

Department of Computing, School of Informatics,
City University London, Northampton Square,
London, England, United Kingdom
{howard.foster.1,g.e.spanoudakis}@city.ac.uk

Abstract. The maturity of IT processes, such as software development, can be and is often certified. Current trends in the IT industry suggest that software systems in the future will be very different from their counterparts today, with an increasing adoption of the Service-Oriented Architecture (SOA) design pattern and the deployment of Software-as-a-Service (SaaS) on Cloud infrastructures. In this talk we discuss some issues surrounding engineering Software Services for Cloud infrastructures and highlight the need for enhanced control, service-level agreement and compliance mechanisms for Software Services. Cloud Infrastructures and Service Mash-ups.

1 Introduction

Aligned with the WESOA workshop theme for this year, this talk discussed the areas of Software Engineering for Cloud, Service-Orientation and Mash-Ups with a focus on safety, security and compliance. Although it is individually challenging to discuss these areas, integrating these together involves further complexity and challenges. The talk specifically aims to highlight the challenges to software engineers, both in engineering discipline at design-time and run-time. We firstly highlighted an interesting case study where architectural decisions in software service design and deployment decisions caused process and resource usage safety issues. Second, we discussed certification for software services and specifically how it enhances a trust mechanism in service discovery and composition. Third, we outlined service policies and constraints, in the form of Service-Level Agreements (SLAs) and how these can be used to monitor services on cloud infrastructures.

2 Safety of Services in Cloud Deployment

Safety for cloud, services and their composition can be viewed from various perspectives. Whilst it is not isolated to service compositions, service design requires some care when deployment is to be made to the cloud and potentially

G. Pallis et al. (Eds.): ICSOC 2011, LNCS 7221, pp. 3–8, 2012.

dynamically changing resource providers and consumers. As an example problem, when enacting a web service orchestration defined using the Web Services Business Process Execution Language (WS-BPEL) we observed various safety property violations. This surprised us considerably as we had previously established that the orchestration was free of such property violations using existing BPEL model checking techniques. In this talk, we described the origins of these violations. They result from a combination of design and deployment decisions, which include the distribution of services across hosts, the choice of synchronisation primitives in the process and the threading configuration of resource containers. We illustrated how model checking can take execution resource constraints into account (see Figure 1). We evaluate the approach by applying it to the above application and are able to demonstrate that a change in allocation of services to hosts is indeed safe, a result that we are able to confirm experimentally in the deployed system. The approach is supported by a tool suite, known as WS-Engineer, providing automated process translation, architecture and model-checking views [2,5,6].

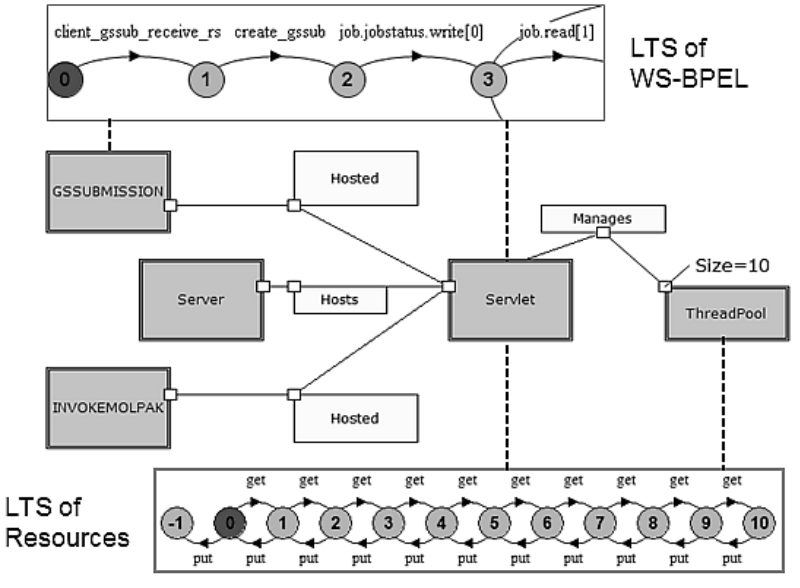

Fig. 1. Example Models for Safety Analysis in Service Deployment

Whilst our focus previously was supporting design-time analysis of service deployment architectures and behavioural models, we also describe how our future work and vision is to provide this at run-time. This involves a collaborative effort between analysis and monitoring. We describe this further in section 5. Our work on analysis however, also raised some interesting points on general service engineering. How do we certify services to be compliant with quality standards?

What are the quality standards that describe this? How are certifications expressed and related between others? What are the safety and security properties we are interested in upholding? These are covered the following parts of the talk.

3 Service Certifications and Cloud Security

Software certification has largely focused on certifying security mechanisms (e.g. ISO/IEC 15408) or product quality metrics (e.g. ISO/IEC 9126*). Our work in certification and assertions for services is part of the EU funded project ASSERT4SOA [8]. ASSERT4SOA aims to provide a general service certification framework, notation and architecture for supporting certificates in service discovery and composition. The ASSERT4SOA language vocabulary is split in to three certification areas: *Evidence-based (ASSERT4SOA-E)*, *Model-based (ASSERT4SOA-M)* and *Ontology-based (ASSERT4SOA-O)*. The three types of certification are used as part of a wider framework to specify and utilise certifications in a service-oriented architecture. As an example usage of this framework, a process for security certificates in service discovery and composition is illustrated in Figure 2.

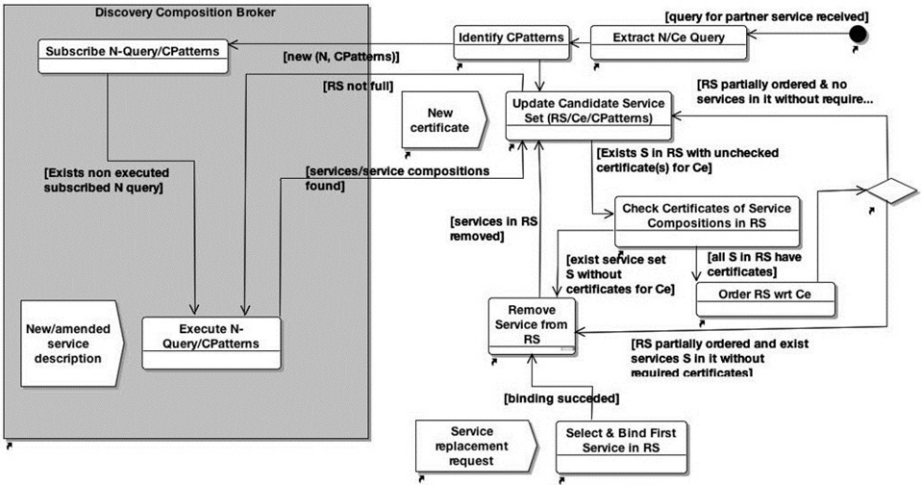

Fig. 2. ASSERT4SOA Framework for Service Composition with Certificates

Note that one particularly challenging area is the trigger from a Service Replacement request. This highlights the dynamic life-cycle of services on "the cloud" and how properties of these services, and the services they rely on, will dynamically change over time and infrastructure changes. Managing these changes and constraining them on Cloud-based *pay-per-usage* model requires an agreement and monitoring architecture.

4 Monitoring Service SLAs in the Cloud

Both Cloud business and infrastructure models emphasise a greater need for accurate levels of service and conformity. With this in mind, specifying and assuring coverage of certain service and infrastructure properties has been undertaken in the EU funded SLA@SOI project [7]. In this project we undertook providing a service monitoring infrastructure with mechanically derived configurations from service-level agreements (SLAs). The architecture for this is illustrated in Figure 3.

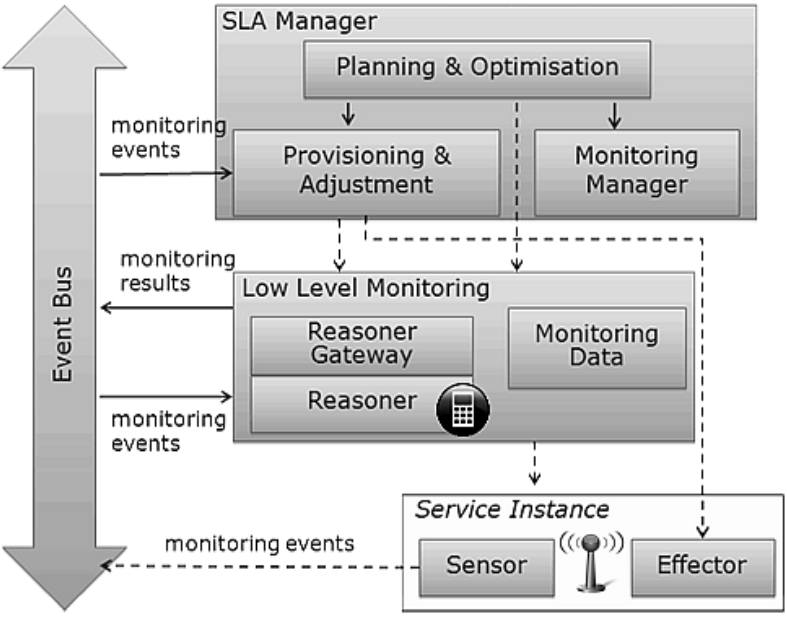

Fig. 3. SLA Monitoring Infrastructure for SOA/Cloud

SLAs are described using the SLA@SOI SLA Model. This includes a vocabulary with terms and constructs for defining complex SLA expressions and constraints. The terms cover a wide range of service and cloud properties, for example, the availability or response time of a service. Both software and infrastructure terms are described, including CPU resource thresholds and virtual machine performance. Certifications provide a way to describe how software meets certain standards. The tests carried out on particular certifications may be drawn from example vocabularies, such as those defined in the SLA@SOI SLA Model. Further details of this work may be found in [3,4].

5 Vision of Service Compliance

Providing both static and dynamic testing tools is a challenge for such techniques discussed previously however we plan to investigate novel yet practical methods for dynamically analysing, monitoring and reacting to violations in such dynamic infrastructures as the Cloud. We discussed mainly a design-time analysis previously, however there is potentially great value to combine service analysis with service monitoring and in the direction of compliance - to ensure certifications are upheld through the software engineering process. Practically, this may be realised through an architecture with checkpoints such as that described by the CBDI [1] and illustrated in Figure 4.

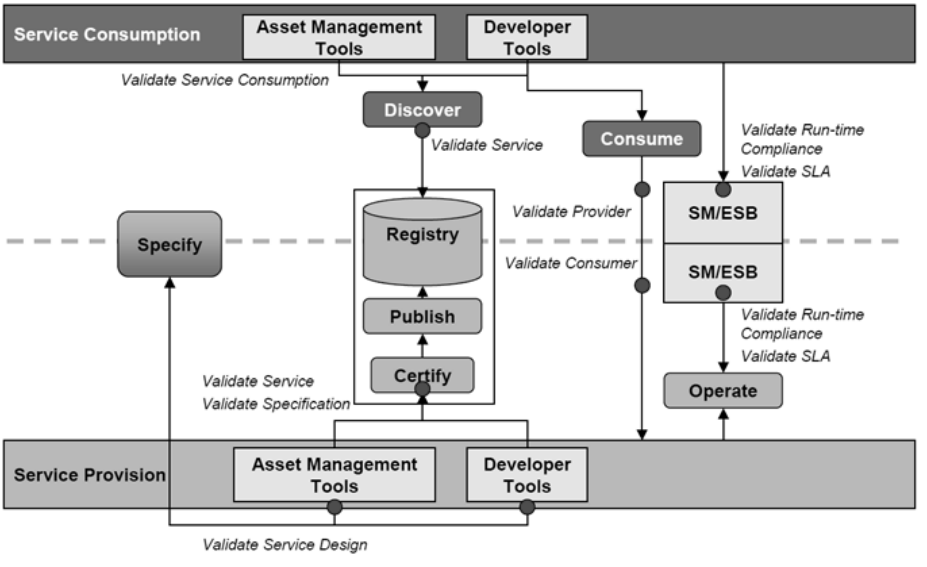

Fig. 4. SOA Architecture with Example Compliance Checkpoints (CBDI 2006)

In summary, we believe that service safety, certification and compliance checking can facilitate a safer cloud of services. As we move towards a more dynamic service-based internet of things, the need for assurance in both providing and consuming services has never been greater. Additionally, new standards of IT compliance may well be delegated to the service engineer to counter such risks as data mis-use and inappropriate service behaviour as a result of ad-hoc service compositions.

Acknowledgements. Our work reported in this keynote has been supported by the EU project ASSERT4SOA - Trustworthy ICT (ICT-2009.1.4).

References

1. Wilkes, L.: Policy Driven Practices for SOA. Presented at the CBDI SOA Seminar (April 2006), http://www.omg.org/news/meetings/workshops
2. Foster, H., Uchitel, S., Magee, J., Kramer, J.: An Integrated Workbench for Model-Based Engineering of Service Compositions. IEEE Transactions on Services Computing (April 2010)
3. Foster, H., Spanoudakis, G.: Dynamic Creation of Service Monitoring Infrastructures. In: Wieder, P., Butler, J.M., Theilmann, W., Yahyapour, R. (eds.) Service Level Agreements for Cloud Computing. Springer, Heidelberg (2011)
4. Foster, H., Spanoudakis, G.: Model-Driven Service Configuration with Formal SLA Decomposition and Selection. In: Proceedings of the 26th ACM Symposium on Applied Computing (SAC), TaiChung, Taiwan (March 2011)
5. Argent-Katwala, A., Clark, A., Foster, H., Gilmore, S., Mayer, P., Tribastone, M.: Safety and Response-Time Analysis of an Automotive Accident Assistance Service. In: Margaria, T., Steffen, B. (eds.) ISoLA 2008. CCIS, vol. 17, pp. 191–205. Springer, Heidelberg (2008)
6. Foster, H., Uchitel, S., Magee, J., Kramer, J.: Model-based Verification of Web Service Compositions. In: IEEE Automated Software Engineering (ASE) 2003, Montreal, Canada (2003)
7. SLA@SOI, A Business-Ready Service-Oriented Infrastructure Empowering The Service Economy in a Flexible and Dependable way, EU Project ICT-2007.1.2, http://sla-at-soi.eu
8. ASSERT4SOA, Advanced Security Service Certificate for SOA, EU Project ICT-2009.1.4, http://assert4soa.eu/