Chapter 15
Considering Quality of a Service in an Intentional Approach

Assia Ait-Ali-Slimane
Université Paris1 Panthéon Sorbonne, France

Manuele Kirsch-Pinheiro
Université Paris1 Panthéon Sorbonne, France

Carine Souveyet
Université Paris1 Panthéon Sorbonne, France

ABSTRACT

The success of service-based applications is based on service technologies such as Web services. Nevertheless, the benefits of the Service-Oriented Architecture (SOA) remain mainly at the software level, since business people are often unable to fully exploit its benefits due to their unfamiliarity with such software level technology. The intentional Service-Oriented Architecture (iSOA) suggests a move from the function-driven SOA to intention-driven SOA in order to provide service description understandable by business practitioners. However, such transposition from business to implementation level should also consider Quality of Service (QoS) aspects. In this paper, we propose modeling the Quality of intentional Service (QoiS) by introducing the quality goals and their qualitative and quantitative evaluation. We also propose populating the intentional service registry of the iSOA architecture with the QoiS description.

CONSIDERING QUALITY OF A SERVICE IN AN INTENTIONAL APPROACH

Service-Oriented Computing (SOC) is the computing paradigm that utilizes services as fundamental elements for developing software applications (Papazoglou et al., 2008). SOC relies on the

Service-Oriented Architecture (SOA) (Alonso et al., 2004) that is a way of reorganizing a portfolio of legacy applications into services that are self-describing computational elements, which are platform independent, accessible through standard interfaces and can be assembled in complex compositions based on standard messaging protocols. Service based applications are considered as support for Business-to-Consumer (B2C) interactions and Business-to-Business (B2B) collaborations.

Services, usually referred as e-services, provide well-defined functionalities that allow users and applications to meet their functional requirements (Casati & Shan, 2001).

SOC is a way of designing a software system that is *function-driven* and it remains at the software level. Consequently, although business people are completely familiar with the notion of service, they are totally unable to fully exploit its benefits, since they are not familiar to such software level technology. SOC technology, such as WSDL (W3C, 2007) and OWL-S (W3C, 2004) is understandable by software professionals, but far to be easily comprehensible by business practitioners. Business practitioners use to reason in terms of business goals.

The *intentional Service-Oriented Architecture approach* (iSOA) (Rolland et al., 2009) suggests a progress from the function-driven SOC to intention-driven SOC in order to provide a service description understandable by business practitioners. The function-driven SOC focuses on a functional view of services, whereas the intention-driven SOC spells out the purpose, the intention behind a service. The main goal of iSOA is to fill the gap between high level business services, referred to *intentional service*, and low level software services. The iSOA approach proposes a higher abstraction level that allows business practitioners to publish, search and compose, in terms of goals and strategies, services that can be executed in the SOA level.

On both level (SOA and iSOA), the service selection remains an important challenge, especially, when a set of services fulfills the same functionality. Among these services, one will be eventually invoked by user, generally depending on a combined QoS evaluation. The QoS can be defined as a set of non functional properties related to software service such as performance, security, accuracy and fault tolerance mechanisms (O'Sullivan et al., 2002; W3C, 2003). On the business level, Fedosseev (2003/2004) describes QoS of business process as a set of qualitative and quantitative characteristics needed to meet the initial requirements of this process. On both levels, QoS stands for non-functional properties that the service provider can ensure and that are (or can be) demanded by the service user.

The QoS plays then an important role in the software service for different reasons (O'Sullivan et al., 2002; Aiello & Giorgini, 2004): (1) a service provider may offer the same functionality with differentiated QoS (for example different prices) and must therefore publish the different qualities for this same functionality; (2) a service requester may decide for a particular service based on its QoS properties; and (3) a service may depend on other services and it needs to be aware of the QoS of the collaborating services. Therefore, QoS becomes a main concern for providers and customers of a service.

Several works in the literature (Zeng et al., 2003; Herssens et al., 2008; Aiello & Giorgini, 2004; Penserini & Mylopoulos, 2005; Ma et al., 2009) propose QoS models. Certain authors (Zeng et al., 2003; Herssens et al., 2008) consider the QoS as a collection of metrics related to non functional properties of services. Zeng et al. (2003) proposes a QoS model to describe concepts of QoS such as execution price and execution duration, whereas Herssens et al. (2008) recommend using the Unified Modeling Language (UML) to enable QoS modeling. Requirement engineering community (Aiello & Giorgini, 2004; Penserini & Mylopoulos, 2005; Ma et al., 2009) propose to reason about non functional requirements, based on the qualitative framework (Mylopoulos et al., 1992), by considering the QoS as soft goal that a service can satisfied.

The expression of the QoS through the different metrics, as defined by (Zeng et al., 2003; Herssens et al., 2008), focuses on the implementation mechanisms of QoS, their computation and their monitoring. Other works such as (Aiello & Giorgini, 2004; Penserini & Mylopoulos, 2005; Ma et al., 2009) focus on the business process. However, an important gap remains between

Related Content

Designing a Framework for Cloud Service Agreement for Cloud Environments
Akashdeep Bhardwajand Sam Goundar (2016). *International Journal of Cloud Applications and Computing (pp. 83-96).*
www.irma-international.org/article/designing-a-framework-for-cloud-service-agreement-for-cloud-environments/173773

Service Flavors: Differentiating Service Offerings in a Services Marketplace
Harshavardhan Jegadeesanand Sundar Balasubramaniam (2010). *Electronic Services: Concepts, Methodologies, Tools and Applications (pp. 50-71).*
www.irma-international.org/chapter/service-flavors-differentiating-service-offerings/43941

International User Interfaces
Barry Jackson (2002). *Internet Management Issues: A Global Perspective (pp. 87-102).*
www.irma-international.org/chapter/international-user-interfaces/24629

Modeling Control Flow in WS-BPEL with Chu Spaces
Xutao Du, Chunxiao Xing, Lizhu Zhouand Ke Han (2011). *International Journal of Information Systems in the Service Sector (pp. 1-21).*
www.irma-international.org/article/modeling-control-flow-bpel-chu/58907

Service Excellence in Marketing for Firms: Strategies and Initiatives
Pratap Chandra Mandal (2022). *International Journal of Service Science, Management, Engineering, and Technology (pp. 1-22).*
www.irma-international.org/article/service-excellence-in-marketing-for-firms/290331