Web Services Compositions Modelling and Choreographies Analysis

Mohsen Rouached, Laboratoire d’InfoRmatique en Image et Systèmes d’information (LIRIS), France
Walid Fdhila, Université Henri Poincaré, France
Claude Godart, Université Henri Poincaré, France

ABSTRACT

In Rouached et al. (2006) and Rouached and Godart (2007) the authors described the semantics of WSBPEL by way of mapping each of the WSBPEL (Arkin et al., 2004) constructs to the EC algebra and building a model of the process behaviour. With these mapping rules, the authors describe a modelling approach of a process defined for a single Web service composition. However, this modelling is limited to a local view and can only be used to model the behaviour of a single process. The authors further the semantic mapping to include Web service composition interactions through modelling Web service conversations and their choreography. This paper elaborates the models to support a view of interacting Web service compositions extending the mapping from WSBPEL to EC, and including Web service interfaces (WSDL) for use in modelling between services. The verification and validation techniques are also exposed while automated induction-based theorem prover is used as verification back-end.

Keywords: Choreography, Orchestration, Semantic Mapping, Verification and Validation, Web Service Composition

INTRODUCTION

The ability to compose complex Web services from a multitude of available component services is one of the most important problems in service-oriented computing paradigm. Web service composition is the ability to aggregate multiple services into a single composite service that would provide a certain functionality, which otherwise cannot be provided by a single service.

While the technology for developing basic services and interconnecting them on a point-to-point basis has attained a certain level of maturity, there remain open challenges when it comes to engineering services that engage in complex interactions that go beyond simple sequences of interactions and responses or involve large numbers of participants.

DOI: 10.4018/jwsr.2010040105
In practice, there are two different (and competing) notions of modeling Web service compositions: orchestration and choreography. Orchestration describes how multiple services can interact by exchanging messages including the business logic and execution order of the interactions from the perspective of a single endpoint (i.e., the orchestrator). It refers to an executable process that may result in a persistent, multi step interaction model where the interactions are always controlled from the point of view of a single entity involved in the process. Choreography, on the other hand, provides a global view of message exchanges and interactions that occur between multiple process endpoints, rather than a single process that is executed by a party. Thus, choreography is more akin to a peer-to-peer (P2P) architecture and offers a means by which the rules of participation for collaboration are clearly defined and agreed upon. Even though there exists competing standards for both the models of composition, WSBPEL for orchestration and WS-CDL (Barros et al., 2005) for choreography, it is widely accepted that both orchestration and choreography can (and should) co-exist within one single environment.

Concerning WS-CDL, as discussed in (Barros et al., 2005), there are several places where its specification is not yet fully developed and a number of known issues remain open. Some issues of a more fundamental or practical nature are difficult to address and are likely to require a significant review of the language’s underlying meta-model and implied techniques. These issues primarily stem from three factors: (i) lack of separation between meta-model and syntax, (ii) lack of direct support for certain categories of use cases and, (iii) lack of comprehensive formal grounding (see Barros et al., 2005).

On the contrary, WSBPEL is quickly emerging as the language of choice for Web service composition. It opens up the possibility of applying a range of formal techniques to the verification of Web services behaviour (see, e.g., Foster et al., 2003; Fu et al., 2004; Pistore et al., 2004). For instance, it is possible to check the internal business process of a participant against the external business protocol that the participant is committed to provide; or, it is possible to verify whether the composition of two or more processes satisfies general properties (such as deadlock freedom) or application-specific constraints (e.g., temporal sequences, limitations on resources). These kinds of verifications are particularly relevant in the distributed and highly dynamic world of Web services, where each partner can autonomously redefine business processes and interaction protocols.

However, one common problem of the different techniques adopted is related to the model used for representing the communications among the Web services. Indeed, the actual mechanism implemented in the existing WSBPEL execution engines is both very complex and implementation dependent. More precisely, WSBPEL processes exchange messages in an asynchronous way; incoming messages go through different layers of software, and hence through multiple queues, before they are actually consumed in the WSBPEL activity; and overpasses are possible among the exchanged messages. On the other hand, the semantics for how to translate the connectivity and communication between activities of the partner processes rather than from a single process focus are not taken into account.

To address these shortcomings, we propose in this paper a semantic framework that provides a foundation for addressing the above limitations by supporting models of service choreography with multiple interacting Web services compositions, from the perspective of a collaborative distributed composition development environment. The process of behaviour analysis moves from a single local process to that of modelling and analyzing the behavior of multiple processes across composition domains. We show also how to translate the connectivity and communication between activities of the partner processes rather than from a single process focus. These may also contain communication actions or dependencies between communication actions that do not appear in any of the service’s behavioral interface(s).
Related Content

Similarity Measures for Substituting Web Services
www.igi-global.com/chapter/similarity-measures-substituting-web-services/66958?camid=4v1a

Efficient Measurement of Service Similarity
www.igi-global.com/article/efficient-measurement-of-service-similarity/90264?camid=4v1a

Web Service versus Distributed Objects: A Case Study of Performance and Interface Design
www.igi-global.com/article/web-service-versus-distributed-objects/3104?camid=4v1a
Optimizing Concurrency Performance of Complex Services in Mobile Environment