Chapter 4

Mediating Message Heterogeneity in Service Compositions:
A Design Model

Prashant Doshi
University of Georgia, USA

Nithya Vembu
University of Georgia, USA

ABSTRACT

Atomic Web Services (WS) may not always be sufficient for service requests. For such cases, several services may have to be assembled to create a new composite service of added functionality and value. Establishing message exchange between related but independently developed Web Services is a key challenge faced during WS composition which has hereto received inadequate attention. One of the challenges lies in resolving the differences in the schema of the messages that are input to and output from the Web Services involved. Data mediation is required to resolve these challenges. This chapter introduces a formal model for data mediation that considers the types and semantics of the message elements. Based on this model, it proposes methods for resolving different kinds of message-level heterogeneity. These methods are evaluated on synthetic and real-world pairs of Web Services, with the ultimate aim of integrating the data mediation techniques presented within WS composition tools.

INTRODUCTION

As reliance of the business and scientific community on services grows, service requests are becoming increasingly sophisticated. Atomic Web Services (WS), whose implementation does not involve other Web Services, are increasingly unlikely to be capable of satisfying such requests. Consequently, WS compositions in which multiple Web Services, both atomic and composite are assembled together to provide the required functionality, is becoming highly relevant.

DOI: 10.4018/978-1-4666-4193-8.ch004
Standardized interfaces, platform transparency, and the Web-based loosely-coupled nature of Web Services make their automated composition feasible. As a result, considerable attention has been directed to the problem of automatically determining the flow of control in a WS composition with the objective of satisfying the functional and non-functional objectives of the composition (Carman et al., 2003; Sirin et al., 2004; Pistore et al., 2005; Agarwal et al., 2005; Zhao & Doshi, 2009). While this is indeed a difficult task, the other key challenge in assembling Web Services to form executable compositions is in ensuring that the participating Web Services can “talk” with each other. In other words, the Web Services should be able to exchange messages and produce the output for the composition correctly.

However, given that Web Services are usually independently developed, their input and output message schemas (i.e., data models) are often heterogeneous. For example, similar data entities may have differing labels and types in the message schemas of different Web Services. A simple example of this may manifest in a StudentRecord WS that utilizes the attribute grade to represent a student’s academic course performance while another student related WS uses the attribute score to represent the same. Nagarajan et al. (2007) list several message-level heterogeneities between Web Services, classified into attribute-level, entity-level and abstraction-level conflicts. Many of these heterogeneities are not new challenges; these have been highlighted previously in different contexts such as matching schemas of federated databases (Litwin & Abdellatif, 1986; Kim et al., 1993; Sheth, 1998; Rahm & Bernstein, 2001). Although by no means exhaustive, the classification does represent a variety of conflicts often found in practice. We use this set of potential heterogeneities between Web Services as our point of departure and mediate them. Specifically, the mediation involves identifying a nontrivial rule or mapping that transforms instances of a concept into instances of a related target concept. For example, grade may be used directly as score or it may need to be transformed into a numeric score.

We begin by noting that our focus is on design-time mediation, which precludes knowing the actual input and output parameters for the Web Services in the composition and thereby its use in the mediation. As a result, the mediation we perform is based on the schema only and represents a conservative approach to the extent of mediation that is possible. The diversity and complexity of potential resolutions make automatic elicitation of mediation rules difficult, and we do not pursue it here.

We take a formal approach toward data mediation between Web Services, and introduce a mathematical model of the message-level mediation problem. This concretely grounds the problem and promotes the development of general approaches toward message-level mediation between Web Services. Our model is sufficiently general in that it captures the different types of heterogeneity listed by Nagarajan et al. (2007) and the requirements for mediating them. The model includes a translator function that defines the set of syntactic and semantic rules that govern the transformation of message schemas. We implement the function in a translator WS whose interface is based on the model and its logic implements the rules. Because multiple types of conflicts may exist in a pair of Web Services participating in a composition, we prescribe a necessary ordering for identifying and resolving the different co-habiting conflicts.

This chapter represents a significant contribution toward addressing multiple types of data mediation between Web Services participating in a composition. We address several data conflicts – many of which are well known – adapted to the context of WS mediation. While the translator WS is well suited for inclusion between two Web Services that need mediation in an orchestration, it may be utilized in choreography as well. We discuss in detail how the translator WS resolves each type of conflict that we consider, and program these conflicts in synthetic Web Services.