Chapter 14
Automating Web Service Composition:
An Ontological Agent Framework

Tamer M. Al Mashat
Cairo University, Egypt

Fatma A. El-Licy
Cairo University, Egypt

Akram I. Salah
Cairo University, Egypt

ABSTRACT

In the Service-Driven Computing paradigm, applications are typically built by composing a set of Web services. Web service composition facilitates rapid development of applications via service reuse and enables the creation of complex services from simpler application services. Research efforts in the area of Web service composition are concerned mainly with two challenges, namely automated service synthesis and verification of the composed Web services. This chapter presents a framework for Web service composition based on semantic specification through OWL-S to establish an ontological agent for automating Web service composition. The semantic description serves to define the planning domain for the agent to automate the composition procedure. A Petri nets model is applied to build a formal representation of the structure and behavior of the service. Finally, AND-OR graph methodology is used to represent the dependences among Web services to select between alternatives based on Quality of Service.

DOI: 10.4018/978-1-4666-6178-3.ch014

INTRODUCTION

The World Wide Web is evolving from being just a collection of pages to being a collection of services that interoperate through the Internet. Services provide a new Web model, in which, sites exchange dynamic information on demand. The Web has become a provider of different kinds of business to customer (B2C) and business to business (B2B) services for governments, organizations, and individuals. With the rapid growth of the Web, its services become more complex and versatile. When a single Web service is not enough to satisfy a required functionality, there exists the potential to compose new services from existing ones.

Service composition, however, is a complicated task to be handled manually. The cause of complexity, in general, is three fold. First, the number of services available over the Web is continuously increasing which complicates the search in the massive Web service repository. Second, existing Web services may be updated and new Web services may be created on the fly, thus, an effective composition system should be able to detect those changes in order to make the decision based on the up to date information. Third, because Web services are developed and specified under a diversity of concept models, there are no agreements on a unique language to define and evaluate a given Web service. This diversity of the service composition complexities motivates the necessity for a semantic understanding of the Web services. These complexities dictate the inevitability of developing an automatic or semi-automatic tool for Web service composition.

Automated Web service composition (WSC) is akin to both the Artificial Intelligence (AI) planning problem and software synthesis problem and draws heavily on the research areas of both (McIlraith & Son, 2002). Software synthesis refers to a process that generates functions, data structures, or entire programs from problem specifications (Hewett, 1996). Automated WSC is achieved through a reasoning system that organizes, combines, and executes Web services, which collectively achieves user's objectives (Pistore, et al. 2005). The reasoning system involves resolving constraints between given Web service inputs, outputs, preconditions, and effects (IOPEs), and typically, the desired outputs and effects (OEs). For example, if one starts with user's goal (some desired outputs and effects), and matches it to the outputs and effects of a Web service (modeled as a process), the result is an instantiation of a process plus descriptions of new goals to be satisfied based on the inputs and preconditions of that process.

The new goals (inputs and preconditions) would hopefully match other processes (outputs and effects), so that composition occurs intrinsically. The constraints between these inputs, outputs, preconditions, and effects dictate the nominated Web services. Web service preconditions and (conditional) effects are not encoded in any existing industrial standard. Yet, they are available as an unambiguous computer-interpretable form in OWL-S (Ouassila, & Zizette, 2011).

Composition problems can be distinguished into two types (Martin et al., 2007): those that involve only information-providing services and those that involve both information-providing and world-altering services. The former requires a rich semantic representation of inputs and outputs (IO), whereas, the latter requires a representation of IOPEs. The findings of matching Web services to a given IOPEs could include several alternatives; therefore, a choice method must be available to elect one of them.

The objective of the introduced ontological agent framework is to provide a higher/complex service by composing a set of Web services when no atomic service meets user requirements. The ontological agent accomplishes this task by manipulating user goals according to the OWL-S service description which defines the planning domain's business process workflow for the agent. Service composition is established by transforming OWL-S process model to Petri nets model