Chapter XII
Web Services Automation

L. Koumakis
Technical University of Crete, Greece
Foundation for Research and Technology–Hellas (FORTH), Greece

V. Moustakis
Technical University of Crete, Greece
Foundation for Research and Technology–Hellas (FORTH), Greece

G. Potamias
Foundation for Research and Technology–Hellas (FORTH), Greece

ABSTRACT

Web services revolutionize use of information systems and to some extent academic research. Put it simply, web services come in small pieces of software, which can be put together to address complex tasks over heterogeneous and distribute data sources. In doing so, services represent an effective solution to isolated software integration. Web services are based on standards; HTML is used as communication protocol layer and XML is used as basic language. Web service composition and orchestration is not trivial. Representation of a human inquiry to an efficient orchestra of services must pass from a translation step, made possible via the use of metadata, which carry real world semantics. To this end, semantic web services denote intention to provide additional information about, and to facilitate, individual service integration and automatic composition. The chapter presents a concrete methodology to support the use of automatic composition of complex semantic web services with natural language. Web service semantics are linked with natural language processing capabilities to empower users to write descriptions in their own language and in the sequel to have these descriptions mapped automatically into a well tuned web service orchestra.
INTRODUCTION

A web service (WS) is defined by the W3C as “a software system designed to support interoperable Machine to Machine interaction over a network. Web services add a new chapter to the success story of semantic web standards. Open standards, like XML, foster universal exchange of information over internet. WS technology adds universal interchangeability and thus universal availability of application logic. Web services represent the second generation of internet tools to connect people to things they are dealing with. They are not connecting people with html web pages; they are connecting their business applications with those of their colleagues, customers, partners and suppliers. Web services could in fact revolutionize the way we develop applications like the internet itself changed our life.

The use of standards has come up as essential part from the first steps of WS technology. Standards are pre-requisite for interoperability. Users want their web services to link and interact with those of their partners and colleagues in a standardized way, yet, personalized to their needs and preferences. The launch of XML opens new avenues for a completely new type of interoperability of software across networks. The desire to use each other’s applications in order to develop new ad-hoc services and appliances is within reach.

Unfortunately semantic web standards cannot solve all the problems that appear in web services. Even though they support interoperability, the development of large and complex domain-specific applications still remains complicated and time consuming.

Interchange availability of web services does not signal automatic pathway to new horizons in designing IT-based business processes. Real applications emerge from a complex and dynamic composition of a number of web services. Almost all business processes span beyond the boundaries of single operation and cross over organizational boundaries.

Choreography and orchestration languages come to fill in that gap and to address the middle layer where atomic services are integrated. An orchestration language (such as Business Process Execution Language) specifies an executable process that involves message exchange with other systems, such that the orchestration designer controls the message exchange sequences. Choreography language specifies interactions and message exchanging through web services in a way to support interoperability and defines legal sequences and interactions among them. Choreography gives the flexibility to the user to select from many different interactions that comply with the language definitions.

The orchestration and the choreography distinctions are based on analogies: orchestration refers to the central control (by the conductor) of the behaviour of a distributed system (the orchestra consisting of many players), while choreography refers to a distributed system (the dancing team) without centralized control.

Technologies related to web services such as SOAP, UDDI and WSDL provide limited support to service discovery, service matching, choreography, orchestration and generally in the process of automated service composition. Furthermore the key actors, the experts of these processes, are not IT experts who nonetheless have to be involved in the design of business processes.

Natural language (NL) by itself is not appropriate for programming, but reference to NL is a key factor for the design of program instructions that are both machine-ready for processing and understandable to humans. For information retrieval and speech recognition systems NL reference is essential and aspect-oriented programming shows how powerful program organizations can be realized based on instructions and structures expressed in NL.