Chapter XI

Web Services: Technology Issues and Foundations

Bernd Amann, University Paris 6, France

Salima Benbernou, University Lyon 1, France

Benjamin Nguyen, University of Versailles, France

Abstract

Unlike traditional applications, which depend upon a tight interconnection of all program elements, Web-service applications are composed of loosely coupled, autonomous, and independent services published on the Web. In this chapter, we first introduce the concept of service-oriented computing (SOC) on the Web and the current standards enabling the definition and publication of Web services. This technology's next evolution is to facilitate the creation and maintenance of Web applications. This can be achieved by exploiting the self-descriptive nature of Web services combined with more powerful models and languages for composing Web services. A second objective of this chapter is to illustrate the complexity of the Web-service composition problem and to provide a representative overview of the existing approaches. The chapter concludes with a short presentation of two research projects exploiting and extending the Web-service paradigm.
Introduction

The possibility to transform existing software components into services and to combine them in a loosely-coupled and flexible way has transformed service-oriented computing (SOC) into the favorite programming and computing paradigm for the development of complex applications on the Web. Unlike traditional applications, which depend upon a tight interconnection of all program elements, service-oriented applications are compositions of autonomous and independent services obtained by using special methods and tools for their design, development, and deployment.

Intuitively, a Web service (or e-service) is an autodescriptive, software-independent, and hardware-independent application that can be exploited using standard Internet networking technologies. From a more technical point of view, a Web service is a dynamic Web resource with a public programming and communication interface by which clients can interact using structured XML (extensible markup language) -based messages conveyed by Internet protocols. The success of this technology is based on the definition of coherent open standards for encoding (XML) and exchanging messages (SOAP, simple object access protocol), as well as for describing (WSDL, Web service definition language) and publishing (UDDI, Universal Description, Discovery and Integration) service operations. Today, these standards confer interoperability and reusability on individual Web services, and the next important step will be to facilitate the creation and maintenance of complex Web-service applications. This evolution can be achieved by exploiting the self-describing nature of Web services combined with more powerful Web-service composition models and languages. The term powerful can be understood in different ways. From a programmer’s point of view, these models and languages should enable the automation of certain programming (e.g., matching service input and output parameters) and administration tasks (e.g., automatically replace a defective service call). Second, they should be able to express and exploit different kinds of static (interface, functionality, and quality) and dynamic (temporal, behavioral) service properties during the design and validation process. One objective of this chapter is to illustrate related technical and scientific issues, their complexity, and a representative overview of the existing solutions.

The rest of this chapter is structured as follows. Next, the chapter shortly presents the industrial and scientific background of the Web-service technology by concentrating on its specificities with respect to other application design and programming infrastructures. Then we describe the problem of building applications with Web services and introduce the main functionalities required for efficient Web-service application development. Next we introduce the general problem of describing a (Web) service and different approaches for service discovery and matchmaking. Web-service composition approaches and languages are studied next, and then the chapter briefly illustrates two research projects exploiting and extending the Web-service paradigm.

Web Services in a Nutshell

Web services are considered a new way of using software components in an information system. Many of the problems that Web services try to solve can be understood by remember-
Pen Testing for Web Applications
[www.igi-global.com/article/pen-testing-web-applications/72989?camid=4v1a](www.igi-global.com/article/pen-testing-web-applications/72989?camid=4v1a)