

Chapter 7.16

Aspect–Oriented Framework for Web Services (AoF4WS): Introduction and Two Example Case Studies

Ghita Kouadri Mostefaoui
Oxford University Computing Laboratory, UK

Zakaria Maamar
Zayed University, UAE

Nanjangud C. Narendra
IBM India Research Lab, India

ABSTRACT

This chapter presents our research initiative known as aspect-oriented framework for Web services (AoF4WS). This initiative looks into the role of aspect-oriented programming in enhancing Web services with nonfunctional properties that are orthogonal to the primary functional properties of Web services, without the need for extensive reprogramming. This enhancement achieves a separation between the functional and nonfunctional aspects of Web services, thereby resulting in easier adaptability and maintainability. We have initially chosen to focus on security and self-healing nonfunctional requirements. The AoF4WS initiative is therefore demonstrated using two projects, SC-WS and SH-

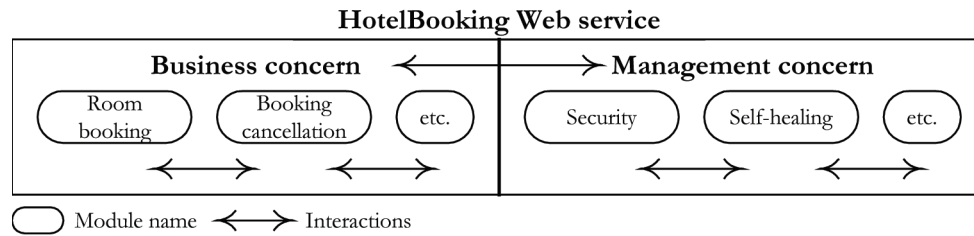
WS, which respectively stand for security concerns of Web services and self-healing Web services. Our contributions are relevant to the design phase in an aspect-oriented software development lifecycle.

INTRODUCTION AND MOTIVATIONS

Web services are an attractive approach for implementing loosely-coupled business processes, which usually spread over companies' boundaries (Ma, 2005). Over the last few years several efforts have been put into the development of standards related to Web services definition, announcement/discovery, and composition, just to cite a few. The dynamic nature of the business world highlights the continuous pressure on businesses to reduce expenses, increase revenues, generate profits, and

DOI: 10.4018/978-1-60566-042-4.ch011

Figure 1. Concern separation in a Web service



remain competitive. This calls for a quick reaction to the market trends, a quick handling of users' needs, a quick adaptation to unforeseen changes, and last but not least, a quick understanding of forthcoming challenges. To boost the acceptance level of Web services by the IT community as the technology of choice when developing flexible processes, Web services need to adapt to changing nonfunctional requirements with minimal reprogramming and minimal maintenance effort, so that they can be kept independent from the core Web services functionality. Security and self-healing are samples of nonfunctional requirements, and we will be highlighting them in this chapter.

Integrating security and self-healing capabilities into Web services calls for a clear separation between “business” and “management” concerns along which a Web service is defined (Figure 1). For this purpose, we adopt an aspect-oriented programming (AOP) approach to specify and implement this separation (Cottenier & Elrad, 2004; El-Manzalawy, 2005). This approach is part of our long-term research initiative known as aspect-oriented framework for Web services (**AoF4WS**). This initiative aims at examining the role of aspects in decoupling various concerns in Web services like security and self-healing. The separation between “business” and “management” sides emphasizes the noninvasive requirement that needs to be taken into consideration during the development cycle of a nonfunctional requirement. The mechanisms related, for instance, to security should be confined into one module and thus, should not scatter over the rest of modules of the

Web service. Figure 1 illustrates the way concern separation occurs in a fictive Web service referred to as HotelBooking. The business side focuses on details directly related to hotel booking, like checking room availability, rate verification, and confirming client reservation. The management side of a Web service gathers all modules, such as security, self-healing, and monitoring that back the operations of this Web service. Constituents of the management side to be implemented as aspects need to be factored out of the core logic of the Web service.

In the following, we describe the two projects that we have chosen as part of the Ao4FWS initiative. The SC-WS project stands for security concerns of Web services and the SH-WS project stands for self-healing Web services. In Section 2 we present some basic definitions necessary for understanding the chapter. A motivating scenario is also presented in this section. SC-WS and SH-WS projects are described in Section 3 and Section 4, respectively. The chapter concludes in Section 5.

BACKGROUND

Some Definitions

Web Service

For the World Wide Web Consortium (W3C), a Web service is a software application identified by a URI, whose interfaces and binding are capable

16 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/chapter/aspect-oriented-framework-web-services/37733

Related Content

An Additively Homomorphic Encryption over Large Message Space

Hu Chen, Yupu Hu, Zhizhu Lian, Huiwen Jia and Xu An Wang (2015). *International Journal of Information Technology and Web Engineering* (pp. 82-102).

www.irma-international.org/article/an-additively-homomorphic-encryption-over-large-message-space/145842

Transforming the Travel Landscape: Smart Contracts in Tourism Management

Jeganathan Gomathi Sankar and Arokiaraj David (2024). *Decentralizing the Online Experience With Web3 Technologies* (pp. 246-266).

www.irma-international.org/chapter/transforming-the-travel-landscape/342268

IoT Architecture

H. Parveen Sultana (2017). *Security Breaches and Threat Prevention in the Internet of Things* (pp. 1-18).

www.irma-international.org/chapter/iot-architecture/177062

Analysis of Quality of Service Routing Algorithms

E. George Dharma Prakash Raj, Sinthu Janita Prakash and S.V. Kasmir Raja (2011). *Web Engineered Applications for Evolving Organizations: Emerging Knowledge* (pp. 159-169).

www.irma-international.org/chapter/analysis-quality-service-routing-algorithms/53059

Analyzing the Long Range Dependence and Object Popularity in Evaluating the Performance of Web Caching

G.P. Sajeev and M.P. Sebastian (2009). *International Journal of Information Technology and Web Engineering* (pp. 25-37).

www.irma-international.org/article/analyzing-long-range-dependence-object/37587