Chapter 3

Quality of Service and Extra–Functional Properties for Web Services: A Model–Driven Approach

Guadalupe Ortiz
University of Cádiz, Spain

Behzad Bordbar
University of Birmingham, UK

ABSTRACT

Service-Oriented Architecture and Web services aim at producing loosely coupled applications which are easier to scale and manage. In addition, loosely coupled systems are less susceptible to errors when modifications and maintenance activities are required. Extending Web services to implement Quality of Service or Extra-functional properties often result in the code that is scattered and tangled in different parts of the main functionality code. This violates decoupleness of the application. In this chapter, we shall present a model-based approach to deal with specification and implementation of Quality of Service and extra-functional properties in a loosely coupled manner.

The presented approach draws on two main software techniques: Model-Driven Architecture, and aspect-oriented programming. The method involves modeling of the Quality of Service and Extra-functional properties in a platform-independent fashion. Then applying model transformation, the platform-independent models are transformed into platform-specific models, and finally into code. The code for Quality of Service and Extra-functional properties are integrated into the system relying on aspect-oriented techniques in a decoupled manner. The presented approach is evaluated with the help of a case study to establish that the approach results in increasing the system’s modularity and thus reducing implementation and maintenance costs.

DOI: 10.4018/978-1-61350-159-7.ch003
INTRODUCTION

Service-Oriented Architecture (SOA) aims at letting distributed applications to communicate, in a platform-independent and loosely coupled manner, providing systems with great flexibility and easier maintenance. Web services have become the best known implementation of SOA, in which various development aspects of the system are maintained based on the use of a wide range of WS standards. Applications developed following Web service standards are loosely coupled; loosely coupled applications are normally easier to scale, manage and extend and less susceptible to errors when modifications are required. Two other significant advantages of adopting Web services are interoperability and the use of Internet standard protocols for communication. Interoperability is achieved via using XML for capturing message and description of the service. Adopting XML allows combining services which are developed on different platforms by different groups of developers. The use of Internet standard protocols, such as http, will permit easier communication through the vast infrastructure. Thus, Web services provide a unique suite of technologies for the integration of business, which allows companies to integrate internal applications, to publish active applications for other companies and to use third-party applications.

On the other hand, the software engineering community is currently moving on to application development based on models rather than technologies. Model-driven development allows us to focus on essential aspects of the system, delaying the decision on which technologies to use in the implementation for later steps of the development. Models may be used in multiple phases of development, from the initial system specification to its testing. Each model will address one or more concerns, independently of the remaining issues involved in the system’s development, thus allowing the separation of the final implementation technology from the business logic achieved by the system. In addition, Model-driven development advocates automated model transformation by using software to create new models from the existing models. Transformations between models enable the automated development of the system from the models themselves; therefore, model definition and transformation become key parts of the process.

Although Web service implementation is properly achieved by development middlewares, Quality of Service (QoS) and Extra-functional Properties (EFP) have not been given due consideration. In particular, in Web services the implementation of QoS – criteria that match the needs of service requestors with those of the service provider – and EFP - functionality which does not perform main but added value system objective - is not carried out in a loosely coupled manner; the functionality related to QoS and EFP is scattered and tangled all over the main functionality code. Consequently, there is a lack of tools for earlier stages of Web service development in which QoS, EFP and WS-* standards integration are considered. The same problem has been studied by component-based and object-oriented communities which are also interested in the separation of concerns. One solution adopted by such communities is to use aspect-oriented techniques, which allow us to encapsulate and modularize transversal concerns in the systems. Using aspects result in improving the decoupleness of software systems and reducing maintenance costs. In our approach we advocate the use of aspect-oriented programming in the implementation of QoS and EFP to create loosely coupled services. So, in a sense our approach brings together Model-Driven Architecture and Aspect-Oriented Software Development (AOSD) to provide good support for software evolution through an appropriate separation of concerns; AOSD has focused on modeling crosscutting concerns whereas MDA concentrates on the explicit separation of platform-independent from platform-specific concerns and the model-driven generation of processes.
Related Content

An Economic Analysis of Cloud: “Software as a Service” (SaaS) Computing and “Virtual Desktop Infrastructure” (VDI) Models
[www.igi-global.com/chapter/economic-analysis-cloud/70147?camid=4v1a](www.igi-global.com/chapter/economic-analysis-cloud/70147?camid=4v1a)

Agent based Resource Allocation Mechanism Focusing Cost Optimization in Cloud Computing

Colored vs. Black Screens or How Color Can Favor Green e-Commerce
[www.igi-global.com/article/colored-black-screens-color-can/53532?camid=4v1a](www.igi-global.com/article/colored-black-screens-color-can/53532?camid=4v1a)

Integrating Product-Service Systems with New Business Models Definition for Manufacturing Industries