Chapter 9

Adaptive and Dynamic Service Compositions in the OSGi Service Platform

Lionel Touseau  
Université de Grenoble, France

Kiev Gama  
Université de Grenoble, France

Didier Donsez  
Université de Grenoble, France

Walter Rudametkin  
Université de Grenoble, France

ABSTRACT

Service-oriented architectures provide a good level of decoupling between the elements that compose an application. Service compositions may take into account that services that take part in the composition can appear and disappear. This is typically not the case when using Web Services. In dynamic environments this uncertain service availability is a recurrent scenario. Applications should be ready to handle that and dynamically adapt their behavior based on the application’s context and the available services. Although typically presented using Web Services, there are also SOAs that use other technologies. In this chapter we provide an overview on some dynamic service oriented platforms, giving special focus on the OSGi Service Platform. Also, we present what principles and mechanisms help to handle dynamicity, and we provide information on the dynamic service-based component models targeting the OSGi platform. These models allow the realization of applications that are adaptive upon dynamic scenarios where service availability is uncertain.

DOI: 10.4018/978-1-61350-159-7.ch009
INTRODUCTION

Software systems are required to become more and more dynamic, flexible and adaptive. There are different factors at runtime that lead to these properties. For example, changing business requirements or application context changes require an application to be easily reconfigurable without stopping its execution. In such highly dynamic scenarios, applications should be able to adapt their behavior autonomously, and be ready to handle failures and unavailability, as well as the apparition of component services, performing the necessary configurations at runtime (Di Nitto, 2008).

Dynamic adaptation is required in various application domains relying on service-oriented computing (SOC) principles (Papazoglou, 2003). In a business context, it is crucial to adapt to market changes, and to take advantage of available service providers to successfully achieve business goals through service composition.

In another emerging context such as ubiquitous computing, also referred to ambient intelligence, systems and applications must adapt to a continuously evolving environment, since they must cope with the characteristics of devices composing those systems. Among these characteristics, mobility is a primary cause of dynamic availability. Data storage and processing units have become smaller and smaller, while radio communication protocols have multiplied (e.g., BlueTooth, 6LoWPAN, WiFi, WiMax, Radio-Frequency IDentification, Near-Field Communication, 3G, ...) and battery life of mobile devices has been considerably extended. All of these factors have resulted in a dissemination of computing resources enabling pervasive and mobile computing, where resources are not static and the services they provide are dynamically available (Weiser, 1993).

SOC has also started being used to provide a flexible and extensible infrastructure for modular applications such as “a la carte” Java EE application servers (e.g., Glassfish or Jonas servers) or plug-in-based applications like the well-known Eclipse IDE. Such systems are likely to evolve at runtime (e.g., the addition, update or removal of new plug-ins and technical services), and thus require the underlying service platform to provide dynamic adaptation capabilities.

Although Service Oriented Architectures are well known through the usage of Web Services, other types of technology take advantage of SOAs and the loose coupling inherent to these architectures to leverage application infrastructure in different scenarios. In this chapter we present some of these technologies, and show how they enable dynamic adaptation, allowing applications to react to changes in the set of services provided in an application’s context.

First, this chapter explains the differences between static service-oriented architectures and dynamic service-oriented architectures, and then it presents the requirements needed for a service platform to enable dynamic adaptation. Next, several dynamic service platforms are described, including the OSGi specification. The second part of this chapter explains the software engineering challenges faced when developing dynamic adaptable service-based applications, and the benefits of service-oriented component models in handling dynamism. It then describes several component models, most of which target the OSGi service platform, which is the main topic of this chapter.

BACKGROUND

Before diving into the depths of dynamic adaptation of service-based applications, here is a background summing up required notions such as the principles of service-oriented computing, the notion of service-based application, and what differs between static and dynamic service composition.
Related Content

Architectural Guidelines and Practical Experiences in the Realization of E-Gov Employment Services
[www.igi-global.com/chapter/architectural-guidelines-practical-experiences-realization/73096?camid=4v1a](www.igi-global.com/chapter/architectural-guidelines-practical-experiences-realization/73096?camid=4v1a)

Trust Management Tools
[www.igi-global.com/chapter/trust-management-tools/30458?camid=4v1a](www.igi-global.com/chapter/trust-management-tools/30458?camid=4v1a)

Gaining Insight into Cognitive Structure Using GALILEO Method: Where is Your Web Site in the Customers’ Cognitive Space?
[www.igi-global.com/chapter/gaining-insight-into-cognitive-structure/44024?camid=4v1a](www.igi-global.com/chapter/gaining-insight-into-cognitive-structure/44024?camid=4v1a)

An Online Intrusion Detection System to Cloud Computing Based on Neucube Algorithms
[www.igi-global.com/article/an-online-intrusion-detection-system-to-cloud-computing-based-on-neucube-algorithms/202391?camid=4v1a](www.igi-global.com/article/an-online-intrusion-detection-system-to-cloud-computing-based-on-neucube-algorithms/202391?camid=4v1a)