Preface

Web services provide a successful way to communicate distributed applications, in a platform independent and loosely coupled manner, providing systems with great flexibility and easier maintenance. In this sense, they are modular applications, which are self-descriptive and can be published, located and invoked from any point on the network. However, even though there are good procedures for the design, development, and management of Web services, there are scopes in which Web service adaptation is required. Service adaptations should be implemented appropriately, so that the loosely coupled nature of Web services is maintained, as well as allowing the implementation’s integration in the whole service lifecycle.

The main aim of this publication is to collect and compile the most representative approaches in current research which tackle the different faces of Web service adaptation. In this regard, readers are able to acquire a panoramic overview of proposed tactics and existing solutions for service adaptation in different development scopes. In this sense, the book has a twofold purpose, not only as background reading on service adaptation, but also as a reference book in the search for adaptation solutions to service software engineering problems.

Therefore the book’s target has been to compile high quality approaches on the different aspects of service adaptation in order to cover the wide spectrum of challenges which emerge in this area under different scenarios.

The target audience of this book covers a wide spectrum of software engineers, from senior and young researchers at the academia to research and development divisions at industry.

The interest of the book for academics is evident: researchers, assistants and professors seek novel and open research fields such as service adaptation; this book will offer this community the chance of having, the most relevant recent literature on the topic in question in a single publication, benefitting from gratifying reading which will provide them both with extensive knowledge of the state of the art of service adaptation as well as on open issues for further research.

On the other hand, industry always keeps an eye on the latest research results, especially on such a dynamic topic as software engineering and moreover on service engineering development. Research divisions from the top ten software industries spend considerable resources on service-related research. Therefore, this book will also be useful for this purpose, not only for them to be aware of what the research in the area is moving towards, but also to let them attempt to include new development ideas in the coming industry software tools.
CHAPTER DESCRIPTION

The book is made up of fourteen chapters which present tactics and solutions for modular and reusable software development in the field of adaptive Web services. These chapters are distributed into five sections: contract-based adaptation and interoperability, context-aware adaptation, adaptation for composition, dynamic adaptation, and device-based future internet adaptation.

Section 1 opens the book with three chapters that show the importance of adaptation and propose tactics to tackle the interoperability issues in service-based systems in an automatic way.

Chapter 1 proposes and specifies an Adaptive Enterprise Service Bus (ESB) Infrastructure which, based on the mediation capabilities provided by ESBs, allows addressing adaptation requirements in service-based systems in an automatic and dynamic way at runtime. The proposed adaptive infrastructure is based on messaging and integration patterns commonly supported in ESB products, so it provides a generic solution which is likely to be applied and implemented in most of these products. It also follows the key ideas of the S-Cube project (http://www.s-cube-network.eu/), exposing adaptation and monitoring capabilities, which can be used by overarching solutions, and leveraging some of its frameworks. This aims to be a step forward in developing dynamic and automatic adaptation capabilities in service-based systems which run over an ESB infrastructure.

Chapter 2 argues that the service-oriented paradigm is a good solution to achieve distributed interoperability, without impairing the capacity of adapting applications to cater for constantly evolving requirements. This goal depends on minimizing coupling between services, so that an adaptation of a service has the lowest possible impact on other services. Therefore, the objective of this chapter is to use structural interoperability and to propose metrics to quantify adaptability based on similarity and decoupling. Thus, this chapter claims to be a contribution to improve the quality of distributed applications and to make their development, maintenance and adaptation an easier task.

Chapter 3 addresses the definition of suitable notions for contract refinement and choreography conformance for services that communicate either synchronously or asynchronously through message queues. Service contracts are specified in a language independent way by means of finite labeled transition systems. This allows the definition of choreography projections in structured operational semantics, which determines that the use of choreography projection is an important step in order to define an appropriate notion of conformance. The notion of service contract presented in this chapter deals only with functional requirements and does not take into accounts non-functional aspects like those related to Service Level Agreement. As regards future research, this chapter proposes to investigate aspects like response time, frequency of client’s invocations, and other quantitative requirements.

Section 2 focuses on context-aware adaptation, where systems must adapt their behaviour at run-time according to the changing conditions, which is tackled in two chapters by using adaptation contracts and event processing.

Chapter 4 presents a framework in order to address the discovery, composition and adaptation processes of pre-existing services by taking context information into account. The main goal of this chapter is to solve, as automatically as possible, mismatch cases which may be given at the different interoperability levels among service interfaces in the field of mobile and pervasive systems. Thus, the framework generates a mediating adaptor based on an adaptation contract obtained by means of a semantic-based matching, which at the same time is generated during the discovery process. This chapter plans future
research directions such as tackling dynamic reconfiguration of services or addressing non-functional requirements at the service level.

Chapter 5 describes an envisaged architecture for context-aware Web services, where context is continuously monitored by a Complex Event Processing engine and service adaptation to context is done through a non-intrusive aspect-oriented implementation. Therefore, this chapter proposes to adapt Web services in multiple scenarios, doing the adaptation in the service-side, thus saving client resources. Aspect-oriented code for adaptation implementation allows the maintenance of the original main functionality service code completely encapsulated and separate from the adaptation code, facilitating not only maintenance of future changes due to new adaptation requirements, but also dynamic adaptations.

Section 3 is made up of three chapters which present different solutions for enhancing the composition of service-based applications by means of technologies like autonomic computing, process mining, and semantic Web.

Chapter 6 tackles the adaptive management of service composition by analyzing how service-oriented systems can self-adapt in order to satisfy non-functional requirements. To do that, the chapter introduces the MAPE-K (Monitor, Analyze, Plan, Execute, and Knowledge) reference model applied to design self-adaptive service-oriented systems. The main focus of the chapter is the Plan phase, since it is the core of the adaptation process. Through a case study, the chapter demonstrates how it is possible to improve the Quality of Service of a service-oriented architecture application that operates in a highly varying execution environment, where component services continuously appear and disappear.

Chapter 7 concerns the application of data mining techniques to a real-life service event log collected by the Vienna Runtime Environment for Service-oriented Computing (VRESCo). The aim of this chapter is to analyze the historical events stored on VRESCo in order to discover software services that are frequently invoked and composed together, i.e., process mining. Two sequential pattern mining algorithms are applied to a real event log provided by VRESCo, with results that show the capability of the approach proposed to find services that are frequently invoked together within the same sequence.

Chapter 8 presents a framework for pervasive service ecosystems, meant to overcome the limitations of standard service-oriented architecture when dealing with situated and adaptive pervasive computing systems, along with an implementation schema based on standard W3C technologies for the Semantic Web, and an application case of crowd steering by public/private displays. Some future works planned are to provide a formal operational model or to develop a methodology for the development of ecosystems, following the AOSE (agent-oriented software engineering) paradigm.

Section 4 tackles the dynamic adaptation issue, where three chapters describe three different frameworks developed to tackle runtime adaptation on both service-based and real-time-constrained systems.

Chapter 9 discusses limitations regarding the dynamic collaboration flexibility of the services involved in the process and introduced a framework for service collaboration that is suitable for cloud computing settings. The chapter also introduces a framework based in techniques for coordination and service matching to achieve dynamic service collaboration through matching goal-oriented service requests with providers that advertise their offerings dynamically. The framework may deal with a large number of providers and requesters working concurrently, which demonstrates the suitability of the approach for large multi-user service coordination environments such as cloud computing.

Chapter 10 presents a new framework for modeling service-based application adaptation triggered by interesting or critical situations based on Situation Action Networks (SAN). SANs are dynamic and can evolve at runtime by using their inherent planning capabilities. Therefore, the framework is based
on a goal model able to track the fulfillment of goals at runtime. The future plans of this chapter include among others the experimentation with search and selection mechanisms that will enable SANs to take advantage of the abstract actions and action pools and support abstraction-based adaptation.

Chapter 11 deals the dynamic adaptation issue from a point of view of the increasing complexity of embedded systems, which are designed to have long life cycles. Thus, this chapter presents an approach to tackle the issue of runtime adaptation on computationally constrained systems, such as embedded systems composed by a set of components, services, or resources, with real-time constraints. The approach focuses on systems with only a single general purpose processor, with the responsibility to run the system and application software. In order to illustrate the framework, its architecture and a case study are described.

Section 5 presents tactics and ideas, in the three last chapters of the book, for addressing adaptive device-based future internet applications, so supporting the interoperability between diverse stakeholders and contents coming from heterogeneous devices.

Chapter 12 proposes a model-driven framework that automates the development of device-aware Web services. This approach allows modelling graphical user interfaces using the notation of the Presentation Modelling Language (PML), whereas the key contribution refers to the transformation of PML models to functional code targeting different platforms encountered on mobile and stationary devices. The chapter considers the preferences of the user when adapting the Web service as a future extension.

Chapter 13 presents how we communicate between different heterogeneous context-aware platforms by using Android and Web services. The main goal is to extend Android to make it adaptable and interoperable. The usefulness of the proposed approach is demonstrated through a concrete case study. As regards future research, this chapter plans to develop a concrete system with context adaption plate forms and data bases in Software-as-a-Service and Cloud environments, and to assess performances of such approach improving it from a security viewpoint.

Chapter 14 finalizes the book by analyzing the opportunities and challenges of adaptive Future Internet (FI) applications. Thus, this chapter discusses the need of rethinking and designing FI applications taking into account aspects beyond the ones considered by current adaptive Web services and service-based applications. There are many questions to be answered, and for each question new ones emerge. Despite all the uncertainties surrounding FI applications, there is at least one certain and incontestable fact: FI applications will have to be engineered explicitly considering adaptation aspects and will have to dynamically adapt to an unprecedented level of changes that may occur during runtime.

**IMPACT FOR THE FIELD OF ADAPTIVE WEB SERVICES**

Service adaptation is a hot topic with many emergent research proposals which to date have not had wide broadcasting, in spite of being worthy of consideration.

This book has been written to facilitate novice readers the understanding of the main issues solved in the proposals as well as providing enough detail in order to hold expert interest.

Chapters along the book provide good quality proposals from both novel and well-established proposals, with solid base and a thorough description of the adaptation problem they deal with and how they tackle it. Challenges and open issues for future work are also an important part of chapter contents, which can encourage prospective interested researchers to work on this relevant area of knowledge.
The editors hope this book will be a helpful reference and inspiring source for research and development activities of software engineers in the field of adaptive Web services at both the academia and the industry.

Guadalupe Ortiz  
*University of Cádiz, Spain*

Javier Cubo  
*University of Málaga, Spain*