

Migrating Legacy Applications:

Challenges in Service Oriented Architecture and Cloud Computing Environments

Anca Daniela Ionita

University "Politehnica" of Bucharest, Romania

Marin Litoiu

York University, Canada

Grace Lewis

Carnegie Mellon Software Engineering Institute, USA

Information Science
REFERENCE

Detailed Table of Contents

Foreword	xii
Preface	xv

Section 1 **Introduction**

Chapter 1

Introduction to the Migration from Legacy Applications to Service Provisioning	1
<i>Anca Daniela Ionita, University "Politehnica" of Bucharest, Romania</i>	

This first chapter provides basic guidelines for understanding the migration to Service-Oriented Architecture and Cloud Computing environments, with references to book chapters where these issues are explored in detail. It analyzes the general context that led to the co-existence of three important needs: changing existing software, preserving legacy, and implementing a gradual migration. It also points out important aspects of migration for three types of targets: SOA, cloud environments, and service-oriented systems in general.

Section 2 **Migrating to SOA Environments**

This section contains literature reviews, research roadmaps, technical solutions, strategies, and practical experiences regarding the migration of legacy applications to SOA environments.

Chapter 2

Research Challenges in the Maintenance and Evolution of Service-Oriented Systems	13
<i>Grace A. Lewis, Carnegie Mellon Software Engineering Institute, USA</i>	
<i>Dennis B. Smith, Carnegie Mellon Software Engineering Institute, USA</i>	

Even if service-oriented architecture represents a validated solution for integrating and leveraging legacy applications, the expectations related to it are sometimes unrealistic, and current research is not focused enough on the primary issues. Therefore, the first chapter of this section presents the research agenda for the maintenance and evolution of service-oriented systems, including tools, techniques, environments, multilingual issues, process reengineering, transition patterns, and runtime monitoring.

Chapter 3

Legacy to SOA Evolution: A Systematic Literature Review	40
---	----

Ravi Khadka, Utrecht University, The Netherlands

Amir Saeidi, Utrecht University, The Netherlands

Andrei Idu, Utrecht University, The Netherlands

Jurriaan Hage, Utrecht University, The Netherlands

Slinger Jansen, Utrecht University, The Netherlands

Taking into account the significant research results reported in the scientific literature during the last decade, the authors present a historical overview of approaches for migrating legacy applications to SOA systems, which bring advantages of portability, decoupling, and a high level of abstraction. Many details of this review are described, starting with its information sources and its strategies, and continuing with the evaluation criteria and the results that led to identification of best practices and open research issues.

Chapter 4

Reengineering and Wrapping Legacy Modules for Reuse as Web Services: Motivation, Method, Tools, and Case Studies	71
--	----

Harry M. Sneed, ANECON GmbH, Austria

While service-oriented architecture is able to support new business processes and the advantages that come with it, users would like to take advantage of the existing large volumes of legacy code and wrap it as Web services to avoid the high costs of developing from scratch. The chapter analyzes both business and technical issues regarding legacy reuse and modernization, inspired from real-life situations. It introduces selection criteria, transformation methods, tools, and lessons learned in three migration projects.

Chapter 5

Service Identification and Specification with SoaML	102
---	-----

Michael Gebhart, Gebhart Quality Analysis (QA) 82, Germany

Based on models of existing legacy systems and their business processes, it is possible to identify and specify the services required for migrating to SOA environments. This chapter uses Service-oriented architecture Modeling Language (SoaML) for describing service interfaces and components. It follows a rigorous approach for creating services, starting from analyzing the requirements and identifying services, and getting into details regarding message and data types.

Chapter 6

The SOA Frontier: Experiences with Three Migration Approaches	126
---	-----

Juan M. Rodriguez, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina

Marco Crasso, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina

Cristian Mateos, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina

Alejandro Zunino, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina

Marcelo Campo, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina

Gonzalo Salvatierra, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina

Many business and government applications are still written in COBOL; meanwhile, the current trend is to adopt service-oriented and Web technologies for integrating large, distributed systems. Given this challenge, migration becomes the right answer, but there is more than one option. This chapter analyzes the performance and assesses the cost implications of three approaches applied for modernizing a government agency: a direct one, wrapping legacy code; an indirect one, redesigning and re-implementing Web services; and a third one, trying to combine the advantages of the first two.

Chapter 7

Model-Driven Software Migration: Process Model, Tool Support, and Application	153
---	-----

Andreas Fuhr, University of Koblenz-Landau, Germany

Andreas Winter, Carl von Ossietzky University, Germany

Uwe Erdmenger, pro et con GmbH, Germany

Tassilo Horn, University of Koblenz-Landau, Germany

Uwe Kaiser, pro et con GmbH, Germany

Volker Riediger, University of Koblenz-Landau, Germany

Werner Teppe, Amadeus Germany GmbH, Germany

It is clear that benefits can be obtained by adopting new technologies and, at the same time, by preserving the software assets validated by practice. Several state-of-the-art methods for reaching this objective are analyzed in the first part of this chapter. Among them, one outlines the horseshoe model, enriched by applying Model-Driven Engineering. This model is used for defining a complete migration process, which takes into account the entire life cycle, followed by presenting a customizable tool suite built for this purpose.

Section 3

Migrating to Cloud Environments

This section presents strategies, solutions, and experiments for migrating existing applications and data to Cloud environments, taking full advantage of the virtualization, scalability, and elasticity potential.

Chapter 8

Moving to SaaS: Building a Migration Strategy from Concept to Deployment	186
--	-----

Leire Orue-Echevarria, TECNALIA, Spain

Juncal Alonso, TECNALIA, Spain

Marisa Escalante, TECNALIA, Spain

Gorka Benguria, TECNALIA, Spain

The transition from off-the-shelf software to Software as a Service entails many business and technical challenges concerning security, scalability, and multi-tenancy, which cannot be solved by classical methods applied for migration to SOA environments. As a consequence, the first chapter of this book section comparatively analyzes the SOA architectural model and the SaaS delivery / business model. The chapter introduces a global strategy for migrating legacy applications to Software as a Service, which takes into account, besides technical issues, business-related phases such as evaluation and decision-making.

Chapter 9

Migration of Data between Cloud and Non-Cloud Datastores	206
--	-----

Shreyansh Bhatt, DA-IICT, India

Sanjay Chaudhary, DA-IICT, India

Minal Bhise, DA-IICT, India

Migrating applications to environments that offer on-demand services also involves migrating data storages, which have to adopt technologies capable of supporting and accessing much larger volumes of data. As a response to this need, the authors provide algorithms for transforming legacy data, based on defining mappings between underlying source data schemas and the target data model, and passing through an RDF/RDFS intermediate model. Such algorithms are demonstrated and evaluated in the chapter.

Chapter 10

Migrating a Legacy Web-Based Document-Analysis Application to Hadoop and HBase: An Experience Report	226
--	-----

Himanshu Vashishtha, University of Alberta, Canada

Michael Smit, University of Alberta, Canada

Eleni Stroulia, University of Alberta, Canada

Besides defining strategies and algorithms, best practices can often be discovered through empirical studies. Applying this principle, the authors of this chapter migrated a Web-based system to the Cloud, adopting two different platforms, Hadoop and HBase. They performed a quantitative analysis of the data storage performance and of the development effort. Criteria for selecting between these alternative solutions are discussed.

Chapter 11

Geographically Distributed Cloud-Based Collaborative Application	248
--	-----

Bogdan Solomon, University of Ottawa, Canada

Dan Ionescu, University of Ottawa, Canada

Cristian Gadea, University of Ottawa, Canada

Marin Litoiu, York University, Canada

An application domain where Cloud environment potential still awaits to be valued is social networking, which undertakes a continuous progression and has to respond to new requirements. An example is sharing online content with colleagues and friends, for which existing products cannot offer enough scalability and synchronization capabilities. The chapter deals with this challenge by performing a gradual migration, first to a cluster of virtualized servers, then to a multi-server application in the Cloud, and finally to a geographically-distributed multi-cloud application.

Section 4

Migrating to Service-Oriented Systems: Frontier Approaches

This section looks at the vision of service-orientation beyond SOA and Cloud, providing an insight on their connections with REST (Representational State Transfer) architectures and Service-Oriented Computing in general.

Chapter 12

Bridging the SOA and REST Architectural Styles	276
--	-----

José C. Delgado, Technical University of Lisbon, Portugal

In distributed systems, interoperability is generally approached by concentrating on one of the following two concerns: behavior and state. The former is addressed in Service Oriented Architecture, while the latter is the focus of the Representational State Transfer (REST) architectural style. This chapter attempts reconciliation between the two concerns, presenting a model that relates resources, services, and processes. It introduces a hierarchy of architectural styles, and presents a structural service style that combines characteristics inherited from SOA and REST.

Chapter 13

Considerations of Adapting Service-Offering Components to RESTful Architectures	303
---	-----

Michael Athanasopoulos, National Technical University of Athens, Greece

Kostas Kontogiannis, National Technical University of Athens, Greece

Chris Brealey, IBM Canada, Canada

This chapter approaches a different kind of migration—adaptation of existing service-oriented systems to the REST style for increasing the potential of scalability and loose coupling. It describes a full adaptation framework that contains: a procedure-oriented API specification, adaptation configuration metadata, the RESTful Service Model, and the Dynamic Behavior Model, together with their generation processes, an intra-service protocol, and various policies.

Chapter 14

Model Driven Integration of Heterogeneous Software Artifacts in Service Oriented	
--	--

Computing	332
-----------------	-----

Eric Simon, Laboratoire Informatique de Grenoble, France

Jacky Estublier, Laboratoire Informatique de Grenoble, France

Migration, as part of system evolution, is based on administration-related actions. If only functionalities are of concern, it is possible to integrate heterogeneous artifacts as services. However, if integration includes application management, like in Cloud Computing, a consistent management of the artifact life cycle is required. A metamodel-based run-time platform is presented in this chapter for supporting such management, which considers multiple life cycle concerns, such as packaging, deployment, and selection.

Compilation of References	361
--	------------

About the Contributors	387
-------------------------------------	------------

Index	396
--------------------	------------