Dragan Gašević
Dragan Djurić
Vladan Devedžić

Model Driven Architecture and Ontology Development

Springer

Model Driven Architecture and Ontology Development

Dragan Gašević · Dragan Djurić
Vladan Devedžić

Model Driven Architecture and Ontology Development

Foreword by Bran Selic

With 153 Figures and 7 Tables

Springer

Authors

Dragan Gašević
School of Interactive Arts
and Technology
Simon Fraser University Surrey
13450 102 Ave.
Surrey, BC V3T 5X3
Canada
dgasevic@sfu.ca
and dgasevic@acm.org

Dragan Djurić
Vladan Devedžić
FON – School of
Business Administration
University of Belgrade
Jove Ilića 154
11000 Belgrade
Serbia and Montenegro
dragandj@gmail.com
devedzic@fon.bg.ac.yu

Library of Congress Control Number: 2006924633

ACM Computing Classification (1998): D.2.10, D.2.11, H.4.0, I.2.4

ISBN-10 3-540-32180-2 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-32180-4 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Typeset by the Authors
Production: LE-TEX Jelonek, Schmidt & Vöckler GbR, Leipzig
Cover design: KünkelLopka Werbeagentur, Heidelberg

Printed on acid-free paper 45/3100/YL - 5 4 3 2 1 0

To our families

Foreword

The first time I paid attention to the term “ontology” was in the late 1980s when I was part of an engineering team that was responsible for defining what we would now call a domain-specific modeling language. In our case, the domain was telecommunications software and the purpose of our language was to give system architects the ability to describe the high-level structure of their software in the most direct and most expressive manner possible.

The team members were all experienced designers with deep knowledge of the domain so that we had no trouble putting together the initial list of key language concepts. We knew that we needed to include standard architectural modeling constructs such as components, ports, connectors, and the like. We also wanted our language to be object-oriented, so notions such as class, objects, and inheritance were added to the list. However, soon after this very promising start, all progress ground to a halt. Somehow, the definition of the seemingly trivial fine-grain details of these constructs kept eluding us despite long, passionate, and occasionally acrimonious discussions that can only be compared to medieval theological debates.

It was our good fortune that at that point we met Professor Doug Skuce of the University of Ottawa. He had a method and a tool that helped us develop an explicit ontology for our domain. From that exercise we learned that our difficulties stemmed from the fact that, although we shared a general intuition for the chosen constructs of our language, there were numerous subtle and *unstated* differences in our individual conceptualizations that were a barrier to mutual understanding. Furthermore, we discovered that certain commonly used terms had multiple meanings – all equally valid – but which we had not differentiated adequately, leading to much confusion. Only after we had defined our ontology, which included semi-formal definitions of all key terms and their relationships, were we able to finish our task successfully.

Ever since I’ve felt that defining a formal domain ontology is a useful and often necessary step in almost any software project. This is because software deals principally with ideas rather than physical artifacts. Whereas the nature of physical artifacts is generally self-evident, this is not the case with conceptual entities, which are products of the mind. As we all know, different minds see the same thing differently.

The definition and application of ontologies for developing software systems is a central theme of this book. However, the book is about much more than that. It explains, in a clear and didactic manner, how a variety of recent buzzword developments in software theory and practice (intelligent agents, Model Driven Architecture, metamodeling, etc.) can be combined, and brings us to the threshold of the next step in the evolution of the World Wide Web: the *Semantic Web*. Like the Internet before it, the Semantic Web promises to introduce a significant and qualitatively new phenomenon into our lives. This is because it endows the network of disparate information that is currently accessible on the Internet with *meaning*. Because this meaning can be gleaned and processed *automatically* by software, the Semantic Web opens up the exciting and awe-inducing possibility of a unified global intelligence accessible to all.

In the first half of the book, the authors navigate deftly through the prolific and highly confusing *gemüscht* of technologies, tools, and standards such as XML, RDF, OWL, MDA, or UML and explain how they relate to each other in the context of the idea of the Semantic Web. They introduce the notion of *modeling spaces,* which provides a conceptually simple yet comprehensive framework for understanding and addressing issues within the domain considered. Using that framework, the second half of the book describes a practical strategy for realizing key elements of the Semantic Web based on existing industry standards.

The book is equally suited to those who merely want to be informed of the relevant technological landscape, to practitioners dealing with concrete problems, and to researchers seeking pointers to potentially fruitful areas of research. The writing is technical yet clear and accessible, and is illustrated throughout with useful and easily digestible examples.

I would also highly recommend this book to sociologists studying the interplay between society and technology. It clearly demonstrates that the core technologies required for constructing the Semantic Web are available and moving forward inexorably. Society must be prepared to deal with something so ripe with potential. We must understand not only how the Semantic Web can be useful but also what dangers lurk within it.

Ottawa, Canada *Bran Selic*

December 2005

Preface

The idea of ontologies emerged in applied artificial intelligence some time ago as a means for sharing knowledge [Gruber, 1993]. Following the development of ontologies and related Web technologies (e.g., HTML and XML), Tim Berners-Lee, Jim Hendler, and Ora Lassila envisioned the next generation of the Web, called the Semantic Web [Berners-Lee et al., 2001]. Being based on ontologies, the Semantic Web has the potential for semantically richer representations of things (e.g., Web pages, applications, and persons) and their relations on the Web, and thus should provide us with more intelligent services. That idea might have initially sounded very futuristic and too enthusiastic, but it has recruited a lot of important players from both academia and industry into very extensive and well-funded research efforts. Today, we have quite impressive results, manifested by standards that have been adopted (RDF and OWL), development frameworks (Jena), best-practice and deployment recommendations, and many applications (e.g. PiggyBank).

Of course, researchers are still facing many challenges in their efforts to accomplish the full vision of the Semantic Web. Probably the first and most important goal is to persuade many industrial developers and software engineers to use and develop ontologies in their everyday practice. However, ontologies rely on well-defined and semantically powerful concepts in artificial intelligence such as description logics, reasoning, and rule-based systems. Since software engineers are largely unfamiliar with these concepts, ontologies have a price that must be paid for the benefits that they provide.

Trying to address the above problems, researchers have started exploring the potential of some widely adopted software engineering tools and methodologies for ontology development. Stephen Cranefield did the pioneering research by proposing that UML, a well-known software modeling language, should be used for ontology development [Cranefield, 2001a]. After him, several researchers have explored further the similarities, differences, and equivalences between UML and ontology languages, as well as the potential of the most recent software engineering initiative called the Model Driven Architecture (MDA), and its accompanying standards (the Meta-Object Facility (MOF) and XML

Metadata Interchange (XMI)) for ontology development [Baclawski et al., 2002a; Djurić et al., 2005a; Falkovych et al., 2003]. This resulted in the initiation of a process for adopting an MDA-based ontology standard by the Object Management Group (OMG), a software engineering standardization consortium [OMG ODM RFP, 2003]. The standard is intended to define the Ontology Definition Metamodel (ODM) using the MOF (used for specifying UML as well), a UML extension (the Ontology UML Profile, or OUP) to allow UML tools to be used to fully develop ontologies, and a set of transformations between the ODM, the OUP, UML, and Semantic Web ontology languages (e.g., RDF(S) and OWL). When completed, the ODM specification is expected to be in the form of an OMG language, like UML and CWM.

In this book we try to fill the gap in the literature covering the subject of applications of the MDA for ontology development on the Semantic Web. Other books cover either the MDA initiative [Kleppe et al., 2003; Mellor et al., 2003b] or the Semantic Web (i.e., ontology development) [Fensel, 2004; Stuckenschmidt & van Harmelen, 2005; Zhong et al., 2003] only. This book gives a comprehensive overview of both themes, with the main emphasis on how we can employ MDA-related standards to develop Semantic Web ontologies. The book is closely related to the recent OMG initiative for the ODM. The book is the first description of that new language.

The book is based on our experience obtained from a series of tutorials entitled "MDA Standards for Ontology Development" that we have given at several international conferences on the Semantic Web (the International Semantic Web Conference and the European Semantic Web Conference) and on software engineering (the International UML Conference and the International Conference of Web Engineering).

Organization and Structure

The book is divided into three parts. Part I covers the basics of both the main topics – ontologies and the MDA. First, Chap. 1 gives a brief overview of the field of knowledge representation in artificial intelligence. Chapters 2 and 3 introduce the main concepts of ontologies, the Semantic Web, standards, applications, tools, and some open research questions. Next, Chap. 4 explains the Model Driven Architecture, and its main standards (the MOF and XMI) and mechanisms (UML profiles). Part I is concluded by Chap. 5 with modeling spaces, a conceptual framework defined to provide an easier understanding of approaches to modeling such as ontologies and MOF-defined modeling languages (UML and the ODM).

Part II is the central part of the book. It starts with Chap. 6, which presents a comprehensive review of several approaches and tools that aim to bridge the gap between ontology development and software engineering methodologies. This chapter also lists the relations between UML and ontology languages. Chapter 7 explains the motivation for the forthcoming OMG ontology development standard for the ODM, and the requirements the standard has to fulfill. Next, Chaps. 8 and 9 describe the current specifications of the OMD and the Ontology UML Profile, respectively. Finally, Chap. 10 analyzes the mappings between MDA-based languages (the ODM and the OUP) and Semantic Web ontology languages.

Part III is dedicated to applications that will support the practical use of languages that conform to the OMG ontology development standard, and to some practical aspects of how to develop ontologies using those MDA-based languages. First, Chap. 11 is a short tutorial showing how to develop ontologies using the OUP in two state-of-the-art UML tools (MagicDraw and Poseidon for UML). Chapter 12 describes an implementation of an ontology-building platform called AIR, developed entirely following MDA principles. Finally, Chap. 13 discusses two examples of ontologies developed using the OUP and MDA standards.

Throughout the book, we use many ontologies, UML and other MDA-based models, and transformations between them. In order to allow you to try them out and use them in practice, we have created a Web page containing supplementary resources. You can reach this Web page at http://www.modelingspaces.org. Besides the resources referred to in the book, this Web page contains the slide handouts of the tutorials that we have given at many international conferences.

Acknowledgments

Some of the content of this book has been previously published by organizations that are not part of the Springer group. We thank these organizations for thier kind permission to use the material for this book, in particular

- IEEE Press
- Rinton Press
- Journal of Educational Technology & Society, and
- Journal of Object Technology.

We are very grateful to Jelena Jovanović, Nenad Krdžavac, and Colin Knight for a huge amount of help in preparing this book and for giving us many useful comments and suggestions. We would also like to thank all members of the GOOD OLD AI research group (http://goodoldai.org.yu) at the University of Belgrade, Serbia and Montenegro for providing us with a great environment and many useful ideas that enormously improved the quality of the book. Dragan Gašević is very grateful to Marek Hatala, Tom Calvert, Griff Richards, and other colleagues from the LORNET research network at Simon Fraser University in Vancouver, Canada, for their great support and understanding.

We thank Bran Selic of IBM Rational Software for agreeing to write the book's foreword. We are also very grateful to Jean Bézivin of the University of Nantes whose research in the filed of model-driven engineering inspired us to start exploring relations between MDA and ontologies. A special mention should be given to Ralf Gerstner of Springer-Verlag, who invited us to write this book and had a lot of patience in working with us. We are very thankful to Evan Wallace, Elisa Kendall, Bob Colomb, Anna Gerber, Dan Chang, Lewis Hart, and other colleagues from the OMG's Ontology PSIG for giving us up-to-date information about the OMG's ODM standardization efforts in the last three years.

Vancouver, Canada	*Dragan Gašević*
Belgrade, Serbia and Montenegro	*Dragan Djurić*
Belgrade, Serbia and Montenegro	*Vladan Devedžić*

March 2006

Contents

Part I Basics

Part II The Model Driven Architecture and Ontologies

Part III Applications