Demystifying OWL for the Enterprise

Synthesis Lectures on the Semantic Web: Theory and Technology

Synthesis Lectures on the Semantic Web: Theory and Technology is edited by Ying Ding of Indiana University and Paul Groth of Elsevier Labs. Whether you call it the Semantic Web, Linked Data, or Web 3.0, a new generation of Web technologies is offering major advances in the evolution of the World Wide Web. As the first generation of this technology transitions out of the laboratory, new research is exploring how the growing Web of Data will change our world. While topics such as ontology-building and logics remain vital, new areas such as the use of semantics in Web search, the linking and use of open data on the Web, and future applications that will be supported by these technologies are becoming important research areas in their own right. Whether they be scientists, engineers or practitioners, Web users increasingly need to understand not just the new technologies of the Semantic Web, but to understand the principles by which those technologies work, and the best practices for assembling systems that integrate the different languages, resources, and functionalities that will be important in keeping the Web the rapidly expanding, and constantly changing, information space that has changed our lives.

Topics to be included:

- Semantic Web Principles from linked-data to ontology design
- Key Semantic Web technologies and algorithms
- Semantic Search and language technologies
- The Emerging "Web of Data" and its use in industry, government and university applications
- Trust, Social networking and collaboration technologies for the Semantic Web
- The economics of Semantic Web application adoption and use

- Publishing and Science on the Semantic Web
- Semantic Web in health care and life sciences

Demystifying OWL for the Enterprise
Michael Uschold
2018

Validating RDF
Jose Emilio Labra Gayo, Eric Prud'hommeaux, Iovka Boneva, and Dimitris Kontokostas
2017

Natural Language Processing for the Semantic Web
Diana Maynard, Kalina Bontcheva, and Isabelle Augenstein
2016

The Epistemology of Intelligent Semantic Web Systems
Mathieu d'Aquin and Enrico Motta
2016

Entity Resolution in the Web of Data
Vassilis Christophides, Vasilis Efthymiou, and Kostas Stefanidis
2015

Library Linked Data in the Cloud: OCLC's Experiments with New Models of Resource Description
Carol Jean Godby, Shenghui Wang, and Jeffrey K. Mixter
2015

Semantic Mining of Social Networks
Jie Tang and Juanzi Li
2015

Social Semantic Web Mining
Tope Omitola, Sebastián A. Ríos, and John G. Breslin
2015

Semantic Breakthrough in Drug Discovery
Bin Chen, Huijun Wang, Ying Ding, and David Wild
2014

Semantics in Mobile Sensing
Zhixian Yan and Dipanjan Chakraborty
2014

Provenance: An Introduction to PROV
Luc Moreau and Paul Groth
2013

Resource-Oriented Architecture Patterns for Webs of Data
Brian Sletten
2013

Aaron Swartz's A Programmable Web: An Unfinished Work
Aaron Swartz
2013

Incentive-Centric Semantic Web Application Engineering
Elena Simperl, Roberta Cuel, and Martin Stein
2013

Publishing and Using Cultural Heritage Linked Data on the Semantic Web
Eero Hyvönen
2012

VIVO: A Semantic Approach to Scholarly Networking and Discovery
Katy Börner, Michael Conlon, Jon Corson-Rikert, and Ying Ding
2012

Linked Data: Evolving the Web into a Global Data Space
Tom Heath and Christian Bizer
2011

Demystifying OWL for the Enterprise
Michael Uschold

ISBN: 978-3-031-79481-0 print
ISBN: 978-3-031-79482-7 ebook
ISBN: 978-3-031-79483-4 hardcover

DOI 10.1007/978-3-031-79482-7

A Publication in the Springer series
SYNTHESIS LECTURES ON THE SEMANTIC WEB: THEORY AND TECHNOLOGY #17

Series Editors: Ying Ding, Indiana University, Paul Groth, Elsevier Labs
Founding Editor: James Hendler

Series ISSN: 2160-4711 Print 2160-472X Electronic

Demystifying OWL for the Enterprise

Michael Uschold
Semantic Arts, Inc.

SYNTHESIS LECTURES ON THE SEMANTIC WEB: THEORY AND TECHNOLOGY #17

ABSTRACT

After a slow incubation period of nearly 15 years, a large and growing number of organizations now have one or more projects using the Semantic Web stack of technologies. The Web Ontology Language (OWL) is an essential ingredient in this stack, and the need for ontologists is increasing faster than the number and variety of available resources for learning OWL. This is especially true for the primary target audience for this book: modelers who want to build OWL ontologies for practical use in enterprise and government settings. The purpose of this book is to speed up the process of learning and mastering OWL. To that end, the focus is on the 30% of OWL that gets used 90% of the time.

Others who may benefit from this book include technically oriented managers, semantic technology developers, undergraduate and post-graduate students, and finally, instructors looking for new ways to explain OWL.

The book unfolds in a spiral manner, starting with the core ideas. Each subsequent cycle reinforces and expands on what has been learned in prior cycles and introduces new related ideas.

Part 1 is a cook's tour of ontology and OWL, giving an informal overview of what things need to be said to build an ontology, followed by a detailed look at how to say them in OWL. This is illustrated using a healthcare example. Part 1 concludes with an explanation of some foundational ideas about meaning and semantics to prepare the reader for subsequent chapters.

Part 2 goes into depth on properties and classes, which are the core of OWL. There are detailed descriptions of the main constructs that you are likely to need in every day modeling, including what inferences are sanctioned. Each is illustrated with real-world examples.

Part 3 explains and illustrates how to put OWL into practice, using examples in healthcare, collateral, and financial transactions. A small ontology is described for each, along with some key inferences. Key limitations of OWL are identified, along with possible workarounds. The final chapter gives a variety of practical tips and guidelines to send the reader on their way.

KEYWORDS

OWL, ontology engineering, data modeling, conceptual modeling, Semantic Web, knowledge graph, enterprise ontology, semantic technology, semantics, reuse, modularity, metadata, resource description framework (RDF), RDF Schema, triples, description logic, knowledge representation

Contents

Foreword by Dave McComb

THE WORLD NEEDS THIS BOOK

The Semantic Web launched in 2001 with Tim Berners-Lee's article in *Scientific American*. By 2004, the W3C had finalized the standardization of the OWL language for modeling ontologies. OWL and the related standards RDF and RDFS enjoyed a brief period of interest, perhaps even hype, in 2007–2009. The interest was short lived.

There were two related reasons that early adopters abandoned the Semantic Web: (1) it was perceived as being too complicated and (2) practitioners didn't understand it.

The "too complicated" rap was partly due to the fact that it is a fairly complex spec, but it was reinforced by the many books, articles, and tutorials that came out at the time. It almost seemed as if the authors intentionally wanted to encourage the perception of a high priesthood, only able to be fathomed by the chosen few.

The "didn't understand it" rap was partly due to the complexity, but persisted because developers and modelers tried to recreate the style of model they were comfortable with. Developers build object-oriented-looking ontologies, and relational database modelers built ontologies that looked a lot like the ER models they were familiar with. Each group was disappointed when they couldn't implement simple constraints, and were frustrated when they finally came up against the "open world assumption."

This is unfortunate because what we have found in the intervening decade is that this modeling language and the technologies that come along with it are the best bet for reversing the siloed mess that most large enterprises deal with on a daily basis.

What Michael has done here is to create a shallow end of this swimming pool. This is a gentle introduction that anyone, even those with the least background in technology or modeling, can easily follow. He introduces and thoroughly explains the 30% of the OWL spec that practicing ontologists use on an everyday basis. He does this without glossing over the things that make OWL and Semantics so special.

Coming away from this book, you will understand the special place that OWL should have in your enterprise datascape. You will understand how OWL fits in with other standards and technologies. You will appreciate how it can simplify your enterprise ontology like no other technology can.

Michael is uniquely suited for the task of writing this book. His Ph.D. was in Artificial Intelligence, specializing in ontology-driven development of ecological simulation software. He was a very early builder and user of ontologies.

After leaving academia he worked in industry designing and applying ontologies for Boeing and Reinvent. He joined Semantic Arts in 2010. In that time he has designed over ten enterprise ontologies and has taught hundreds of budding ontologists in the subtleties of this technology.

This combination of theoretical background coupled with pragmatic experience is, to the best of my knowledge, unequaled.

It has been a pleasure working with Michael at Semantic Arts these seven-plus years.

Enjoy this book, and welcome to the next generation of enterprise information systems!

Dave McComb

Foreword by Mark A. Musen

ONTOLOGIES ARE EVERYWHERE

We can't order merchandize online, stream a movie, search the Web, or access social media without interacting with software that uses ontologies. Most of the software that surrounds us and that we often take for granted has at its core ontologies—almost always taken for granted—that characterize the merchandize, the movies, the websites, the users, and everything else that the software needs to compute about. Whether they are actually called ontologies, or product catalogs, or knowledge graphs, data structures that capture models of the entities in the world with which a system interacts are critical components of modern computing technologies.

But ontologies sure are hard to build. Understanding how the entities in some application area might be modeled, how they might interrelate with one another, and how they might be captured in software is really difficult. The problem is exacerbated because the standard computer language for representing ontologies—OWL—is complex and often nonintuitive. OWL causes all kinds of problems for new ontology engineers. In OWL, it's easy to infer that left is right and that up is down, unless you are extremely careful. An OWL ontology that states that an opera is a play in which all the words are sung will also classify as an opera a pantomime—a play that has no words at all! If you're not vigilant, OWL might tell you that a gall bladder is a golf club or that a toothbrush is a sonnet.

Why do we put up with this nonsense? We use OWL because it has many useful properties that allow us to understand the implications of our modeling choices, enabling us to have more confidence that we have modeled things correctly. We also use OWL because it has become an international standard. Before OWL became a recommendation of the World Wide Web Consortium, there was no prevailing language for encoding ontologies, no easy way to integrate ontologies, and few widely used tools for building ontologies. That chaos disappeared with the advent of OWL. Standardization simplified many practical aspects of ontology engineering and allowed the ontology-development community to share ontology content, ontology-engineering systems, and best practices for ontology engineering on a broad scale.

The Protégé ontology editor developed by my group at Stanford University was the first widely used tool for building ontologies in OWL, and it remains the only open-source platform that supports OWL-based ontology development that is in common use. Over the years, we've continued to enhance Protégé with additional features that, we believe, help users to deal with

many of the complexities of ontology engineering in OWL. But none of these features overcomes the basic problem that description logics in general, and OWL in particular, have elements that are unintuitive and hard to learn.

I teach about OWL at Stanford, and students who are new to the language always seem to end up building ontologies that classify toothbrushes as sonnets, scratching their heads trying to figure out why. OWL's "open world assumption" and its somewhat arcane methods for defining the characteristics of the entities in a model are initially hard to grasp. Students are confused that language developers would choose to make things so difficult and complicated. They persevere, however, because they appreciate the importance of ontology engineering in the development of many modern software systems and the critical role that ontologies play in many modern professional activities, particularly in the sciences. Students still struggle, both because the syntax is difficult and because the implications of their modeling choices are often hard to predict. As an instructor, I have been frustrated that there is no easy way to teach students the basic components of knowledge modeling in OWL other than to sit them down in front of an ontology editor such as Protégé and ask them to represent the essential features of a toothbrush or a golf club. To date, all the writings about OWL have come from scientists in the description-logic community whose main objectives have been completeness and accuracy rather than pedagogy and understanding. The literature is thus one written by experts for experts, leaving novice learners to stare helplessly at pages of complex equations and at unpronounceable abbreviations such as SHOIN and SHROIQ, always set in ridiculous fonts.

This book is different. It clearly "demystifies" OWL by distilling the language to its very basic features and by presenting clear, easy-to-follow examples. The emphasis is on communicating clearly the core elements of the language, rather than on expansiveness and logical rigor. This volume is an important contribution, coming at an important time, as ontologies enter the mainstream of software engineering and are no longer in the exclusive domain of highly specialized experts.

This book marks an important transition. At last, there is recognition that ontologies are often being constructed in the course of routine software development. It is now apparent that there is a need for more traditional software engineers to be able to create ontologies and to render them in OWL, the representation language that has now entered standard usage. This is a natural evolution in the trend toward translation of human knowledge into computable forms, enabling new technologies to interact with people and to communicate human ideas in novel ways. Most people don't think critically about the ontologies that allow them to find the products that they want online or that suggest the content in which they might be interested on social media; their interactions with these ontologies come naturally and implicitly. Similarly, the software engineers who build these semantically aware systems should not get hung up on the complexities or enigmatic properties of OWL; their work to model and represent ontologies should come just as instinctively.

It's about time that OWL became demystified. The next generation of intelligent software systems depends on it.

Mark A. Musen
Palo Alto, California

Preface

WHY OWL?

The current state of information technology in the modern enterprise has been described as a "Software Wasteland".[1] There are countless silos where each application has its own database and its own database schema with consequent duplication and high costs of integration and change. There are a few root causes.

First, there is no mechanism for breaking up a data schema into modules that can be re-purposed and reused across multiple databases. Monolithic data models and the lack of reusability increase the cost of change. Second, there is no way to uniquely identify data or schema elements globally across databases; this results in high integration costs. Finally, and perhaps most importantly, there are no widely used technologies and practices for representing the meaning of the data and schema elements as they evolve. Conceptual models may exist, but they are rarely kept up to date.

After a slow incubation period of nearly 15 years, the modern enterprise is waking up to the value of the Semantic Web stack of technologies, which has addressed all 3 of the above root causes.

The meaning of the data in a semantic application is defined using the Web Ontology Language (OWL). An OWL ontology is a model that represents the subject matter of the data (e.g., healthcare or electrical products) that will reside in triple-store databases that will be used by one or more related applications. OWL is built on the Resource Description Framework (RDF) which is a knowledge graph language based on triples. The support for globally unique identifiers is baked into RDF and thus OWL.

OWL is an essential ingredient in the semantic technology stack that continues to grow and evolve.

Leading-edge enterprises are building their own enterprise ontologies and enterprise knowledge graphs.[2,3] The technology stack includes a graph query language analogous to SQL called SPARQL for querying triple stores. More recent additions to the stack include R2RML for converting data from relational databases into triples and SHACL for representing constraints and

1 *Software Wasteland* by D. McComb, https://technicspub.com/software_wasteland/,
2 Exploiting Linked Data and Knowledge Graphs in Large Organisations http://www.springer.com/us/book/9783319456522.
3 Linking Enterprise Data. http://www.springer.com/us/book/9781441976642.

other details that are pertinent to specific applications. These are separate important tools for use in conjunction with OWL ontologies, and are beyond the scope of this book.

Finally, there are ample industry-scale tools provided by vendors that support these standards. The time is now to step out of the software wasteland.

WHY THIS BOOK?

It's not easy to learn OWL on your own, even if you have a Ph.D. in artificial intelligence and training in formal logic. I found that out the hard way in 2002, when I was tasked with learning a precursor to OWL at a research lab at The Boeing Company. The purpose of this book is to dramatically speed up that learning process for others.

In 2010, I joined Semantic Arts as a senior ontology consultant and have been teaching OWL and using it to build industrial ontologies for the past seven-plus years. As few as five years ago, one of our clients told us not to use the "O" word (ontology)—because it would scare people. Back then, hardly anyone in a typical enterprise knew much about ontology, and there were few if any projects going on. That has dramatically changed in the past five years.

The need for ontologists is growing faster than the number and variety of available resources for learning OWL, especially from an industry perspective. This book differs from others available at the time of writing in that it is focused primarily on the needs of the modeler in an industrial context. I take a demand-pull approach, only introducing an OWL construct when the need for it arises to meet a modeling need. I focus on the 30% of OWL that gets used 90% of the time. Finally, I use examples from real-world industrial ontologies created in my day-to-day work.

The material in the first two chapters of this book has been presented five times in the past four years as a half-day tutorial. Venues included the Semantic Technology Conference, International Semantic Web Conference, Semantic Technology for Intelligence, Defense and Security, Data Architecture Summit, and Enterprise Data World. Some of the material has been folded into the Designing and Building Business Ontology class that I co-teach for Semantic Arts. The success I had with this material inspired me to expand it into a book.

TARGET AUDIENCE

This book is a gentle but thorough introduction to the most important parts of OWL. The only prerequisites for this book are an interest in modeling and a knack for thinking logically. The primary audience consists of modelers who want to build OWL ontologies for practical use in enterprise and government settings. For them, I drive most of the learning from real-world examples and avoid unnecessarily technical language. Secondary target audiences include the following.

- *Intermediate to expert modelers in any setting* having some familiarity with OWL who wish to deepen their understanding and see things from a fresh perspective.
- *Technically oriented managers* who want to know about ontology and OWL to better interact with their technical people.
- *Undergraduates and post-graduates* who want to understand OWL from a practical enterprise perspective.
- *Instructors* who are looking for new ways to explain OWL.
- *Semantic technology developers* who want a better understanding of the OWL ontologies that they code to.

OVERVIEW OF THIS BOOK

The book unfolds in a spiral manner. In the first cycle, I describe the core ideas. Each subsequent cycle reinforces and expands on what has been learned in prior cycles and introduces new related ideas. The book is divided into three parts.

Part 1: Introducing OWL

This is a cook's tour of ontology and OWL, giving an informal overview of what things need to be said to build an ontology, followed by a detailed look at how to say them in OWL. This is illustrated using a healthcare example. I conclude by explaining some foundational ideas about meaning and semantics to prepare for going deeper in the next section.

Part 2: Going into Depth: Properties, Classes, and Inference

Everything to do with building an OWL ontology revolves around properties and classes. I give detailed descriptions of the main constructs that you are likely to need in everyday modeling, including what inferences are sanctioned.

Part 3: Using OWL in Practice

Using examples in healthcare, collateral, and financial transactions, we put into practice what we have learned so far. For each, I describe a model and show some inferences. Next, I identify some key limitations of OWL and possible workarounds. I conclude with a variety of practical tips and guidelines to send you on your way.

STYLE

Per common practice, most of this book is written using the editorial "we." At times, "we" will refer to the collective shared views and experiences of myself and my ontologist colleagues. "I" is used to express a personal perspective that may not be shared by my colleagues.

EXERCISES

In a number of places throughout the book you will find exercises. Answers may be found in Appendix A.4.

Acknowledgments

I first acknowledge those who made this book possible. I am deeply grateful to Dave McComb for providing an opportunity to build so many ontologies in such a wide variety of industries; and for the many insightful ideas he has shared over the years. My views as an ontologist have been greatly influenced by him. Dave also gave insightful feedback on a later draft of this book.

My parents' greatest gift was to encourage me and my eight siblings to pursue *our* own dreams, not theirs. A special thanks to Professor Richard Uschold (my Dad) who shared with me his unique gift for explaining complex technical ideas. Alan Bundy helped me through some tough times in graduate school and taught me how to organize a major writing project.

I have Ying Ding to thank for asking me a couple years ago: "Have you ever thought about writing a book?" I thank Larry Raymore for planting a seed that got me started. I'm grateful to Michael Morgan who agreed to publish this book and has been a pleasure to work with.

My views on OWL itself and how it is most effectively taught have been influenced by the numerous ontologists and modelers that I have worked with over the years. These include Semantic Arts colleagues: Andi Engelstad,[4] Dan Carey, Dave McComb, Mark Ouska, Mark Wallace, Simon Robe, and Ted Hills. It also includes ontologists authoring the Financial Industry Business Ontology (FIBO). These include David Newman, Dean Allemang, Elisa Kendall, Mike Bennet, and Pete Rivett. FIBO was also a source of some of the examples I used in this book. Thanks to Simon Robe also for permission to use his visual OWL syntax.

My hero reviewers were Dalia Varanka and Mark Ouska who read every word, sometimes more than once, and turned around drafts in record time. In addition, I received a lot of useful feedback on different chapters from Andrea Engelstad, Andrea Zachary, Bobbin Teegarden, Dave Plummer, Matt Johnson, and Pete Rivett. Andrea Zachary made a major contribution to the index.

Finally, I am deeply grateful to Mary VonRanker, for listening to and supporting me in numerous ways. She was a very effective sounding board for ideas about the book.

[4] All lists of names are in alphabetical order.