A Semantic Web Primer

second edition

Grigoris Antoniou

and

Frank van Harmelen

The MIT Press
Cambridge, Massachusetts
London, England

Series Foreword

The traditional view of information systems as tailor-made, cost-intensive database applications is changing rapidly. The change is fueled partly by a maturing software industry, which is making greater use of off-the-shelf generic components and standard software solutions, and partly by the onslaught of the information revolution. In turn, this change has resulted in a new set of demands for information services that are homogeneous in their presentation and interaction patterns, open in their software architecture, and global in their scope. The demands have come mostly from application domains such as e-commerce and banking, manufacturing (including the software industry itself), training, education, and environmental management, to mention just a few.

Future information systems will have to support smooth interaction with a large variety of independent multivendor data sources and legacy applications, running on heterogeneous platforms and distributed information networks. Metadata will play a crucial role in describing the contents of such data sources and in facilitating their integration.

As well, a greater variety of community-oriented interaction patterns will have to be supported by next-generation information systems. Such interactions may involve navigation, querying and retrieval, and will have to be combined with personalized notification, annotation, and profiling mechanisms. Such interactions will also have to be intelligently interfaced with application software, and will need to be dynamically integrated into customized and highly connected cooperative environments. Moreover, the massive investments in information resources, by governments and businesses alike, call for specific measures that ensure security, privacy, and accuracy of their contents.

All these are challenges for the next generation of information systems. We call such systems *cooperative information systems,* and they are the focus of this series.

In lay terms, cooperative information systems are serving a diverse mix of demands characterized by *content—community—commerce*. These demands are originating in current trends for off-the-shelf software solutions, such as enterprise resource planning and e-commerce systems.

A major challenge in building cooperative information systems is to develop technologies that permit continuous enhancement and evolution of current massive investments in information resources and systems. Such technologies must offer an appropriate infrastructure that supports not only development but also evolution of software.

Early research results on cooperative information systems are becoming the core technology for community-oriented information portals or gateways. An information gateway provides a "one-stop-shopping" place for a wide range of information resources and services, thereby creating a loyal user community.

The research advances that will lead to cooperative information systems will not come from any single research area within the field of information technology. Database and knowledge-based systems, distributed systems, groupware, and graphical user interfaces have all matured as technologies. While further enhancements for individual technologies are desirable, the greatest leverage for technological advancement is expected to come from their evolution into a seamless technology for building and managing cooperative information systems.

The MIT Press Cooperative Information Systems series will cover this area through textbooks, and research editions intended for the researcher and the professional who wishes to remain up-to-date on current developments and future trends.

The series will include three types of books:

- Textbooks or resource books intended for upper-level undergraduate or graduate level courses

- Research monographs, which collect and summarize research results and development experiences over a number of years

- Edited volumes, including collections of papers on a particular topic

Data in a data source are useful because they model some part of the real world, its subject matter (or *application*, or *domain of discourse*). The problem of *data semantics* is establishing and maintaining the correspondence between a data source, hereafter a *model*, and its intended subject matter. The model may be a database storing data about employees in a company, a database schema describing parts, projects, and suppliers, a Web site presenting information about a university, or a plain text file describing the battle of Waterloo. The problem has been with us since the development of the first databases. However, the problem remained under control as long as the operational environment of a database remained closed and relatively stable. In such a setting, the meaning of the data was factored out from the database proper and entrusted to the small group of regular users and application programs.

The advent of the Web has changed all that. Databases today are made available, in some form, on the Web where users, application programs, and uses are open-ended and ever changing. In such a setting, the semantics of the data has to be made available along with the data. For human users, this is done through an appropriate choice of presentation format. For application programs, however, this semantics has to be provided in a formal and machine-processable form. Hence the call for the Semantic Web.[1]

Not surprisingly, this call by Tim Berners-Lee has received tremendous attention by researchers and practitioners alike. There is now an International Semantic Web Conference series,[2] a Semantic Web Journal published by Elsevier,[3] as well as industrial committees that are looking at the first generation of standards for the Semantic Web.

The current book constitutes a timely publication, given the fast-moving nature of Semantic Web concepts, technologies, and standards. The book offers a gentle introduction to Semantic Web concepts, including XML, DTDs, and XML schemas, RDF and RDFS, OWL, logic, and inference. Throughout, the book includes examples and applications to illustrate the use of concepts.

We are pleased to include this book on the Semantic Web in the series on Cooperative Information Systems. We hope that readers will find it interesting, insightful, and useful.

John Mylopoulos
jm@cs.toronto.edu
Dept. of Computer Science
University of Toronto
Toronto, Ontario
Canada

Michael Papazoglou
M.P.Papazoglou@kub.nl
INFOLAB
P.O. Box 90153
LE Tilburg
The Netherlands

[1] Tim Berners-Lee and Mark Fischetti, *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. San Francisco: HarperCollins, 1999.

[2] <http://iswc.semanticweb.org>.

[3] <http://www.semanticwebjournal.org>.

Preface

The World Wide Web (WWW) has changed the way people communicat with each other, how information is disseminated and retrieved, and hov business is conducted. The term *Semantic Web* comprises techniques tha promise to dramatically improve the current WWW and its use. This book i about this emerging technology.

The success of each book should be judged against the authors' aims. Thi is an introductory textbook about the Semantic Web. Its main use will be t serve as the basis for university courses about the Semantic Web. It can als be used for self-study by anyone who wishes to learn about Semantic Wel technologies.

The question arises whether there is a need for a textbook, given that al information is available online. We think there is a need because on the Wel there are too many sources of varying quality and too much information Some information is valid, some outdated, some wrong, and most source talk about obscure details. Anyone who is a newcomer and wishes to lear something about the Semantic Web, or who wishes to set up a course on th Semantic Web, is faced with these problems. This book is meant to help out

A textbook must be selective in the topics it covers. Particularly in a fielc as fast developing as this, a textbook should concentrate on fundamenta aspects that can reasonably be expected to remain relevant some time int the future. But, of course, authors always have their personal bias.

Even for the topics covered, this book is not meant to be a reference worl that describes every small detail. Long books have already been written o certain topics, such as XML. And there is no need for a reference work i the Semantic Web area because all definitions and manuals are available online. Instead, we concentrate on the main ideas and techniques and provid enough detail to enable readers to engage with the material constructivel and to build applications of their own.

That way readers will be equipped with sufficient knowledge to easily ge the remaining details from other sources. In fact, an annotated list of references is found at the end of each chapter.

Preface to the Second Edition

The reception of the first edition of this book showed that there was a real need for a book with this profile. The book is in use in dozens of courses worldwide and has been translated into Japanese, Spanish, Chinese and Korean.

The Semantic Web area has seen rapid development since the first publication of our book. New elements have appeared in the Semantic Web language stack, new application areas have emerged, and new tools are being produced. This has prompted us to produce a second edition with a substantial number of updates and changes. In brief, this second edition has the following new elements:

- All known bugs and errata have been fixed (notably the RDF chapter (chapter 3) contained some embarrassing errors).

- The RDF chapter now discusses SPARQL as the RDF query language (with SPARQL going for W3C recommendation in the near future, and already receiving widespread implementation support).

- The OWL chapter (chapter 4) now discusses OWL DLP, a newly identified fragment of the language with a number of interesting practical and theoretical properties.

- In the light of rapid developments in this area, the chapter on rules (chapter 5) has been revised and discusses the SWRL language as well as OWL DLP.

- New example applications have been added to chapter 6.

- The discussion of web services in chapter 6 has been revised and is now based on OWL-S.

- The final outlook chapter (chapter 8) has been entirely rewritten to reflect the advancements in the state of the art, to capture a number of currently ongoing discussions, and to list the most challenging issues facing the Semantic Web.

We have also started to maintain a Web site with material to support th use of this book: <http://www.semanticwebprimer.org>. The Web site con tains slides for each chapter, to be used for teaching, online versions of cod fragments in the book, and links to material for further reading.

Acknowledgments

We thank Jeen Broekstra, Michel Klein, and Marta Sabou for pioneerin much of this material in our course on Web-based knowledge representatio at the Free University in Amsterdam; Annette ten Teije, Zharko Aleksovsk and Wouter Jansweijer for critically reading early versions of the manuscript and Lynda Hardman and Jacco van Ossenbruggen for spotting errors in th RDF chapter.

We thank Christoph Grimmer and Peter Koenig for proofreading parts o the book and assisting with the creation of the figures and with LaTeX pro cessing.

For the second edition of this book, the following people generously con tributed material: Jeen Broekstra wrote section 3.9 on SPARQL; Peter Mik and Michel Klein wrote section 6.3 on their openacademia system; some o the text on the Bibster system in section 6.4 was donated by Peter Haase fron his Ph.D. thesis; and some of the text on OWL-S was donated by Marta Sabo from her Ph.D. thesis.

Also, we wish to thank the MIT Press people for their assistance with the fi nal preparation of the manuscript, and Christopher Manning for his LATEX 2_{ε} macros.

1 The Semantic Web Vision

1.1 Today's Web

The World Wide Web has changed the way people communicate with eacl other and the way business is conducted. It lies at the heart of a revolution that is currently transforming the developed world toward a knowledg economy and, more broadly speaking, to a knowledge society.

This development has also changed the way we think of computers. Originally they were used for computing numerical calculations. Currently thei predominant use is for information processing, typical applications bein database systems, text processing, and games. At present there is a transition of focus toward the view of computers as entry points to the informatio highways.

Most of today's Web content is suitable for human consumption. Eve Web content that is generated automatically from databases is usuall presented without the original structural information found in databases Typical uses of the Web today involve people's seeking and making use o information, searching for and getting in touch with other people, reviewing catalogs of online stores and ordering products by filling out forms, anc viewing adult material.

These activities are not particularly well supported by software tools Apart from the existence of links that establish connections between documents, the main valuable, indeed indispensable, tools are search engines.

Keyword-based search engines such as Yahoo and Google are the mair tools for using today's Web. It is clear that the Web would not have becom the huge success it is, were it not for search engines. However, there ar serious problems associated with their use:

- High recall, low precision. Even if the main relevant pages are retrieved they are of little use if another 28,758 mildly relevant or irrelevant documents are also retrieved. Too much can easily become as bad as too little.

- **Low or no recall**. Often it happens that we don't get any relevant answer for our request, or that important and relevant pages are not retrieved. Although low recall is a less frequent problem with current search engines, it does occur.

- **Results are highly sensitive to vocabulary**. Often our initial keywords do not get the results we want; in these cases the relevant documents use different terminology from the original query. This is unsatisfactory because semantically similar queries should return similar results.

- **Results are single Web pages.** If we need information that is spread over various documents, we must initiate several queries to collect the relevant documents, and then we must manually extract the partial information and put it together.

Interestingly, despite improvements in search engine technology, the difficulties remain essentially the same. It seems that the amount of Web content outpaces technological progress.

But even if a search is successful, it is the person who must browse selected documents to extract the information he is looking for. That is, there is not much support for retrieving the information, a very time-consuming activity. Therefore, the term *information retrieval*, used in association with search engines, is somewhat misleading; *location finder* might be a more appropriate term. Also, results of Web searches are not readily accessible by other software tools; search engines are often isolated applications.

The main obstacle to providing better support to Web users is that, at present, the meaning of Web content is not *machine-accessible*. Of course, there are tools that can retrieve texts, split them into parts, check the spelling, count their words. But when it comes to *interpreting* sentences and extracting useful information for users, the capabilities of current software are still very limited. It is simply difficult to distinguish the meaning of

> I am a professor of computer science.

from

> I am a professor of computer science, you may think. Well, . . .

Using text processing, how can the current situation be improved? One solution is to use the content as it is represented today and to develop increasingly sophisticated techniques based on artificial intelligence and computational linguistics. This approach has been followed for some time now, but despite some advances the task still appears too ambitious.

An alternative approach is to represent Web content in a form that is more easily machine-processable[1] and to use intelligent techniques to take advantage of these representations. We refer to this plan of revolutionizing the Web as the *Semantic Web* initiative. It is important to understand that the Semantic Web will not be a new global information highway parallel to the existing World Wide Web; instead it will gradually evolve out of the existing Web.

The Semantic Web is propagated by the World Wide Web Consortium (W3C), an international standardization body for the Web. The driving force of the Semantic Web initiative is Tim Berners-Lee, the very person who invented the WWW in the late 1980s. He expects from this initiative the realization of his original vision of the Web, a vision where the meaning of information played a far more important role than it does in today's Web.

The development of the Semantic Web has a lot of industry momentum and governments are investing heavily. The U.S. government has established the DARPA Agent Markup Language (DAML) Project, and the Semantic Web is among the key action lines of the European Union's Sixth Framework Programme.

1.2 From Today's Web to the Semantic Web: Examples

1.2.1 Knowledge Management

Knowledge management concerns itself with acquiring, accessing, and maintaining knowledge within an organization. It has emerged as a key activity of large businesses because they view internal knowledge as an intellectual asset from which they can draw greater productivity, create new value, and increase their competitiveness. Knowledge management is particularly important for international organizations with geographically dispersed departments.

[1] In the literature the term *machine-understandable* is used quite often. We believe it is the wrong word because it gives the wrong impression. It is not necessary for intelligent agents to *understand* information; it is sufficient for them to process information effectively, which sometimes causes people to think the machine really understands.

Most information is currently available in a weakly structured form, for example, text, audio, and video. From the knowledge management perspective, the current technology suffers from limitations in the following areas:

- Searching information. Companies usually depend on keyword-based search engines, the limitations of which we have outlined.

- Extracting information. Human time and effort are required to browse the retrieved documents for relevant information. Current intelligent agents are unable to carry out this task in a satisfactory fashion.

- Maintaining information. Currently there are problems, such as inconsistencies in terminology and failure to remove outdated information.

- Uncovering information. New knowledge implicitly existing in corporate databases is extracted using data mining. However, this task is still difficult for distributed, weakly structured collections of documents.

- Viewing information. Often it is desirable to restrict access to certain information to certain groups of employees. "Views," which hide certain information, are known from the area of databases but are hard to realize over an intranet (or the Web).

The aim of the Semantic Web is to allow much more advanced knowledge management systems:

- Knowledge will be organized in conceptual spaces according to its meaning.

- Automated tools will support maintenance by checking for inconsistencies and extracting new knowledge.

- Keyword-based search will be replaced by query answering: requested knowledge will be retrieved, extracted, and presented in a human-friendly way.

- Query answering over several documents will be supported.

- Defining who may view certain parts of information (even parts of documents) will be possible.

1.2.2 Business-to-Consumer Electronic Commerce

Business-to-consumer (B2C) electronic commerce is the predominant commercial experience of Web users. A typical scenario involves a user's visiting one or several online shops, browsing their offers, selecting and ordering products.

Ideally, a user would collect information about prices, terms, and conditions (such as availability) of all, or at least all major, online shops and then proceed to select the best offer. But manual browsing is too time-consuming to be conducted on this scale. Typically a user will visit one or a very few online stores before making a decision.

To alleviate this situation, tools for shopping around on the Web are available in the form of shopbots, software agents that visit several shops, extract product and price information, and compile a market overview. Their functionality is provided by wrappers, programs that extract information from an online store. One wrapper per store must be developed. This approach suffers from several drawbacks.

The information is extracted from the online store site through keyword search and other means of textual analysis. This process makes use of assumptions about the proximity of certain pieces of information (for example the price is indicated by the word *price* followed by the symbol $ followed by a positive number). This heuristic approach is error-prone; it is not always guaranteed to work. Because of these difficulties only limited information is extracted. For example, shipping expenses, delivery times, restrictions on the destination country, level of security, and privacy policies are typically not extracted. But all these factors may be significant for the user's decision making. In addition, programming wrappers is time-consuming, and changes in the online store outfit require costly reprogramming.

The Semantic Web will allow the development of software agents that can *interpret* the product information and the terms of service:

- Pricing and product information will be extracted correctly, and delivery and privacy policies will be interpreted and compared to the user requirements.

- Additional information about the reputation of online shops will be retrieved from other sources, for example, independent rating agencies or consumer bodies.

- The low-level programming of wrappers will become obsolete.

- More sophisticated shopping agents will be able to conduct automated negotiations, on the buyer's behalf, with shop agents.

1.2.3 Business-to-Business Electronic Commerce

Most users associate the commercial part of the Web with B2C e-commerce, but the greatest economic promise of all online technologies lies in the area of business-to-business (B2B) e-commerce.

Traditionally businesses have exchanged their data using the Electronic Data Interchange (EDI) approach. However this technology is complicated and understood only by experts. It is difficult to program and maintain, and it is error-prone. Each B2B communication requires separate programming, so such communications are costly. Finally, EDI is an isolated technology. The interchanged data cannot be easily integrated with other business applications.

The Internet appears to be an ideal infrastructure for business-to-business communication. Businesses have increasingly been looking at Internet-based solutions, and new business models such as *B2B portals* have emerged. Still, B2B e-commerce is hampered by the lack of standards. HTML (hypertext markup language) is too weak to support the outlined activities effectively: it provides neither the structure nor the semantics of information. The new standard of XML is a big improvement but can still support communications only in cases where there is a priori agreement on the vocabulary to be used and on its meaning. XSLT

The realization of the Semantic Web will allow businesses to enter partnerships without much overhead. Differences in terminology will be resolved using standard *abstract domain models,* and data will be interchanged using translation services. Auctioning, negotiations, and drafting contracts will be carried out automatically (or semiautomatically) by software agents.

1.2.4 Wikis

Currently, the use of the WWW is expanded by tools that enable the active participation of Web users. Some consider this development revolutionary and have given it a name: Web 2.0.

Part of this direction involves *wikis,* collections of Web pages that allow users to add content (usually structured text and hypertext links) via a browser interface. Wiki systems allow for collaborative knowledge creation because they give users almost complete freedom to add and change information without ownership of content, access restrictions, or rigid workflow Wiki systems are used for a variety of purposes, including the following:

- Development of bodies of knowledge in a community effort, with contr butions from a wide range of users. The best-known result is the genera purpose Wikipedia.

- Knowledge management of an activity or a project. Examples are brair storming and exchanging ideas, coordinating activities, and exchangin records of meetings.

While it is still early to talk about drawbacks and limitations of this techno ogy, wiki systems can definitely benefit from the use of semantic technolc gies. The main idea is to make the inherent structure of a wiki, given b the linking between pages, accessible to machines beyond mere navigatior This can be done by enriching structured text and untyped hyperlinks wit semantic annotations referring to an underlying model of the knowledg captured by the wiki. For example, a hyperlink from *Knossos* to *Heraklio* could be annotated with information *is located in*. This information coul then be used for context-specific presentation of pages, advanced querying and consistency verification.

1.2.5 Personal Agents: A Future Scenario

The following scenario illustrates functionalities that can be implemente based on Semantic Web technologies.

Michael had just had a minor car accident and was feeling some neck pair His primary care physician suggested a series of physical therapy sessions Michael asked his Semantic Web agent to work out some possibilities.

The agent retrieved details of the recommended therapy from the doctor' agent and looked up the list of therapists maintained by Michael's healtl insurance company. The agent checked for those located within a radius of 1 km from Michael's office or home, and looked up their reputation accordin to trusted rating services. Then it tried to match available appointment time with Michael's calendar. In a few minutes the agent returned two proposals Unfortunately, Michael was not happy with either of them. One therapis had offered appointments in two weeks' time; for the other Michael woulc have to drive during rush hour. Therefore, Michael decided to set stricte time constraints and asked the agent to try again

A few minutes later the agent came back with an alternative: a therapist with a good reputation who had available appointments starting in two days. However, there were a few minor problems. Some of Michael's less important work appointments would have to be rescheduled. The agent offered to make arrangements if this solution were adopted. Also, the therapist was not listed on the insurer's site because he charged more than the insurer's maximum coverage. The agent had found his name from an independent list of therapists and had already checked that Michael was entitled to the insurer's maximum coverage, according to the insurer's policy. It had also negotiated with the therapist's agent a special discount. The therapist had only recently decided to charge more than average and was keen to find new patients.

Michael was happy with the recommendation because he would have to pay only a few dollars extra. However, because he had installed the Semantic Web agent a few days ago, he asked it for explanations of some of its assertions: how was the therapist's reputation established, why was it necessary for Michael to reschedule some of his work appointments, how was the price negotiation conducted? The agent provided appropriate information.

Michael was satisfied. His new Semantic Web agent was going to make his busy life easier. He asked the agent to take all necessary steps to finalize the task.

1.3 Semantic Web Technologies

The scenarios outlined in section 1.2 are not science fiction; they do not require revolutionary scientific progress to be achieved. We can reasonably claim that the challenge is an engineering and technology adoption rather than a scientific one: partial solutions to all important parts of the problem exist. At present, the greatest needs are in the areas of integration, standardization, development of tools, and adoption by users. But, of course, further technological progress will lead to a more advanced Semantic Web than can, in principle, be achieved today.

In the following sections we outline a few technologies that are necessary for achieving the functionalities previously outlined.

1.3.1 Explicit Metadata

Currently, Web content is formatted for human readers rather than programs. HTML is the predominant language in which Web pages are written (directly or using tools). A portion of a typical Web page of a physical therapist mig look like this:

```
<h1>Agilitas Physiotherapy Centre</h1>
Welcome to the Agilitas Physiotherapy Centre home page.
Do you feel pain? Have you had an injury? Let our staff
Lisa Davenport, Kelly Townsend (our lovely secretary)
and Steve Matthews take care of your body and soul.

<h2>Consultation hours</h2>
Mon 11am - 7pm<br>
Tue 11am - 7pm<br>
Wed 3pm - 7pm<br>
Thu 11am - 7pm<br>
Fri 11am - 3pm<p>
But note that we do not offer consultation
during the weeks of the
<a href=". . .">State Of Origin</a> games.
```

For people the information is presented in a satisfactory way, but machin will have their problems. Keyword-based searches will identify the wor *physiotherapy* and *consultation hours*. And an intelligent agent might even l able to identify the personnel of the center. But it will have trouble disti guishing the therapists from the secretary, and even more trouble finding tl exact consultation hours (for which it would have to follow the link to tl State Of Origin games to find when they take place).

The Semantic Web approach to solving these problems is not the deve opment of superintelligent agents. Instead it proposes to attack the proble from the Web page side. If HTML is replaced by more appropriate language then the Web pages could carry their content on their sleeve. In additic to containing formatting information aimed at producing a document fo human readers, they could contain information about their content. In ou example, there might be information such as

```
<company>
  <treatmentOffered>Physiotherapy</treatmentOffered>
  <companyName>Agilitas Physiotherapy Centre</companyName>
  <staff>
    <therapist>Lisa Davenport</therapist>
    <therapist>Steve Matthews</therapist>
    <secretary>Kelly Townsend</secretary>
  </staff>
</company>
```

This representation is far more easily processable by machines. The term *metadata* refers to such information: data about data. Metadata capture part of the *meaning* of data, thus the term *semantic* in Semantic Web.

In our example scenarios in section 1.2 there seemed to be no barriers in the access to information in Web pages: therapy details, calendars and appointments, prices and product descriptions, it seemed like all this information could be directly retrieved from existing Web content. But, as we explained, this will not happen using text-based manipulation of information but rather by taking advantage of machine-processable metadata.

As with the current development of Web pages, users will not have to be computer science experts to develop Web pages; they will be able to use tools for this purpose. Still, the question remains why users should care, why they should abandon HTML for Semantic Web languages. Perhaps we can give an optimistic answer if we compare the situation today to the beginnings of the Web. The first users decided to adopt HTML because it had been adopted as a standard and they were expecting benefits from being early adopters. Others followed when more and better Web tools became available. And soon HTML was a universally accepted standard.

Similarly, we are currently observing the early adoption of XML. While not sufficient in itself for the realization of the Semantic Web vision, XML is an important first step. Early users, perhaps some large organizations interested in knowledge management and B2B e-commerce, will adopt XML and RDF, the current Semantic Web-related W3C standards. And the momentum will lead to more and more tool vendors' and end users' adopting the technology.

This will be a decisive step in the Semantic Web venture, but it is also a challenge. As we mentioned, the greatest current challenge is not scientific but rather one of technology adoption.

1.3.2 Ontologies

The term *ontology* originates from philosophy. In that context, it is used as the name of a subfield of philosophy, namely, the study of the nature of existence (the literal translation of the Greek word $O\nu\tau o\lambda o\gamma\iota\alpha$), the branch of metaphysics concerned with identifying, in the most general terms, the kinds of things that actually exist, and how to describe them. For example, the observation that the world is made up of specific objects that can be grouped

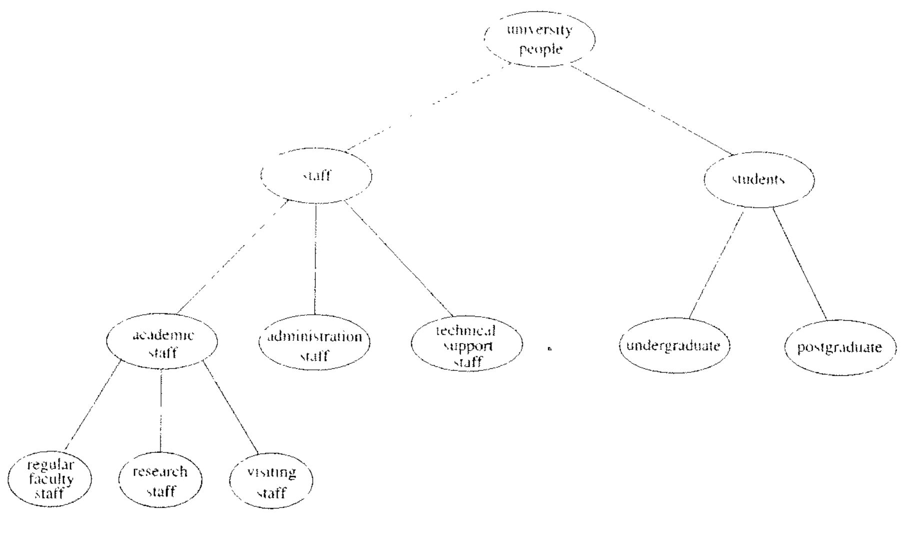

Figure 1.1 A hierarchy

into abstract classes based on shared properties is a typical ontological commitment.

However, in more recent years, *ontology* has become one of the many words hijacked by computer science and given a specific technical meaning that is rather different from the original one. Instead of "ontology" we now speak of *"an* ontology." For our purposes, we will use T. R. Gruber's definition, later refined by R. Studer: *An ontology is an explicit and formal specification of a conceptualization.*

In general, an ontology describes formally a domain of discourse. Typically, an ontology consists of a finite list of terms and the relationships between these terms. The *terms* denote important *concepts* (*classes* of objects) of the domain. For example, in a university setting, staff members, students, courses, lecture theaters, and disciplines are some important concepts.

The *relationships* typically include hierarchies of classes. A hierarchy specifies a class C to be a subclass of another class C' if every object in C is also included in C'. For example, all faculty are staff members. Figure 1.1 shows a hierarchy for the university domain.

Apart from subclass relationships, ontologies may include information such as

- **properties** (X teaches Y),
- **value restrictions** (only faculty members may teach courses),
- **disjointness statements** (faculty and general staff are disjoint),
- **specifications of logical relationships between objects** (every department must include at least ten faculty members).

In the context of the Web, ontologies provide *a shared understanding of a domain.* Such a shared understanding is necessary to overcome differences in terminology. One application's zip code may be the same as another application's area code. Another problem is that two applications may use the same term with different meanings. In university A, a course may refer to a degree (like computer science), while in university B it may mean a single subject (CS 101). Such differences can be overcome by mapping the particular terminology to a shared ontology or by defining direct mappings between the ontologies. In either case, it is easy to see that ontologies support semantic interoperability .

Ontologies are useful for the organization and navigation of Web sites. Many Web sites today expose on the left-hand side of the page the top levels of a concept hierarchy of terms. The user may click on one of them to expand the subcategories.

Also, ontologies are useful for improving the accuracy of Web searches. The search engines can look for pages that refer to a precise *concept* in an ontology instead of collecting all pages in which certain, generally ambiguous, keywords occur. In this way, differences in terminology between Web pages and the queries can be overcome.

In addition, Web searches can exploit generalization/specialization information. If a query fails to find any relevant documents, the search engine may suggest to the user a more general query. It is even conceivable for the engine to run such queries proactively to reduce the reaction time in case the user adopts a suggestion. Or if too many answers are retrieved, the search engine may suggest to the user some specializations.

In Artificial Intelligence (AI) there is a long tradition of developing and using ontology languages. It is a foundation Semantic Web research can build upon. At present, the most important ontology languages for the Web are the following:

- RDF is a data model for objects ("resources") and relations between them;

it provides a simple semantics for this data model; and these data model can be represented in an XML syntax.

- RDF Schema is a vocabulary description language for describing properties and classes of RDF resources, with a semantics for generalizatio hierarchies of such properties and classes.

- OWL is a richer vocabulary description language for describing properties and classes, such as relations between classes (e.g., disjointness), cardinality (e.g., "exactly one"), equality, richer typing of properties, characteristics of properties (e.g., symmetry), and enumerated classes.

1.3.3 Logic

Logic is the discipline that studies the principles of reasoning; it goes back t Aristotle. In general, logic offers, first, *formal languages* for expressing knowledge. Second, logic provides us with *well-understood formal semantics*: i most logics, the meaning of sentences is defined without the need to operationalize the knowledge. Often we speak of declarative knowledge: w describe *what* holds without caring about *how* it can be deduced.

And third, automated reasoners can deduce (infer) conclusions from th given knowledge, thus making implicit knowledge explicit. Such reasoners have been studied extensively in AI. Here is an example of an inference Suppose we know that all professors are faculty members, that all facult members are staff members, and that Michael is a professor. In predicat logic the information is expressed as follows:

$$prof(X) \rightarrow faculty(X)$$

$$faculty(X) \rightarrow staff(X)$$

$$prof(michael)$$

Then we can deduce the following:

$$faculty(michael)$$

$$staff(michael)$$

$$prof(X) \rightarrow staff(X)$$

Note that this example involves knowledge typically found in ontologies. Thus logic can be used to uncover ontological knowledge that is implicitly given. By doing so, it can also help uncover unexpected relationships and inconsistencies.

But logic is more general than ontologies. It can also be used by intelligent agents for making decisions and selecting courses of action. For example, a shop agent may decide to grant a discount to a customer based on the rule

$$loyalCustomer(X) \rightarrow discount(X, 5\%)$$

where the loyalty of customers is determined from data stored in the corporate database. Generally there is a trade-off between expressive power and computational efficiency. The more expressive a logic is, the more computationally expensive it becomes to draw conclusions. And drawing certain conclusions may become impossible if noncomputability barriers are encountered. Luckily, most knowledge relevant to the Semantic Web seems to be of a relatively restricted form. For example, our previous examples involved *rules* of the form, "If conditions, then conclusion," where conditions and conclusion are simple statements, and only finitely many objects needed to be considered. This subset of logic, called **Horn logic, is tractable and supported by efficient reasoning tools.**

An important advantage of logic is that it can provide *explanations* for conclusions: the series of inference steps can be retraced. Moreover AI researchers have developed ways of presenting an explanation in a human-friendly way, by organizing a proof as a natural deduction and by grouping a number of low-level inference steps into metasteps that a person will typically consider a single proof step. Ultimately an explanation will trace an answer back to a given set of facts and the inference rules used.

Explanations are important for the Semantic Web because they increase users' confidence in Semantic Web agents (see the physiotherapy example in section 1.2.5). Tim Berners-Lee speaks of an "Oh yeah?" button that would ask for an explanation.

Explanations will also be necessary for activities between agents. While some agents will be able to draw logical conclusions, others will only have the capability to *validate proofs*, that is, to check whether a claim made by another agent is substantiated. Here is a simple example. Suppose agent 1, representing an online shop, sends a message "You owe me $80" (not in natural language, of course, but in a formal, machine-processable language) to agent 2, representing a person. Then agent 2 might ask for an explanation, and agent 1 might respond with a sequence of the form

Web log of a purchase over $80

Proof of delivery (for example, tracking number of UPS)

Rule from the shop's terms and conditions:

$$purchase(X, Item) \wedge price(Item, Price) \wedge delivered(Item, X)$$
$$\rightarrow owes(X, Price)$$

Thus facts will typically be traced to some Web addresses (the trust of which will be verifiable by agents), and the rules may be a part of a shared commerce ontology or the policy of the online shop.

For logic to be useful on the Web it must be usable in conjunction with other data, and it must be machine-processable as well. Therefore, there is ongoing work on representing logical knowledge and proofs in Web languages. Initial approaches work at the level of XML, but in the future rules and proofs will need to be represented at the level of RDF and ontology languages, such as DAML+OIL and OWL.

1.3.4 Agents

Agents are pieces of software that work autonomously and proactively. Conceptually they evolved out of the concepts of object-oriented programming and component-based software development.

A personal agent on the Semantic Web (figure 1.2) will receive some tasks and preferences from the person, seek information from Web sources, communicate with other agents, compare information about user requirements and preferences, select certain choices, and give answers to the user. An example of such an agent is Michael's private agent in the physiotherapy example of section 1.2.5.

It should be noted that agents will not replace human users on the Semantic Web, nor will they necessarily make decisions. In many, if not most, cases their role will be to collect and organize information, and present choices for the users to select from, as Michael's personal agent did in offering a selection between the two best solutions it could find, or as a travel agent does that looks for travel offers to fit a person's given preferences.

Semantic Web agents will make use of all the technologies we have outlined:

- Metadata will be used to identify and extract information from Web sources.

- Ontologies will be used to assist in Web searches, to interpret retrieved information, and to communicate with other agents.

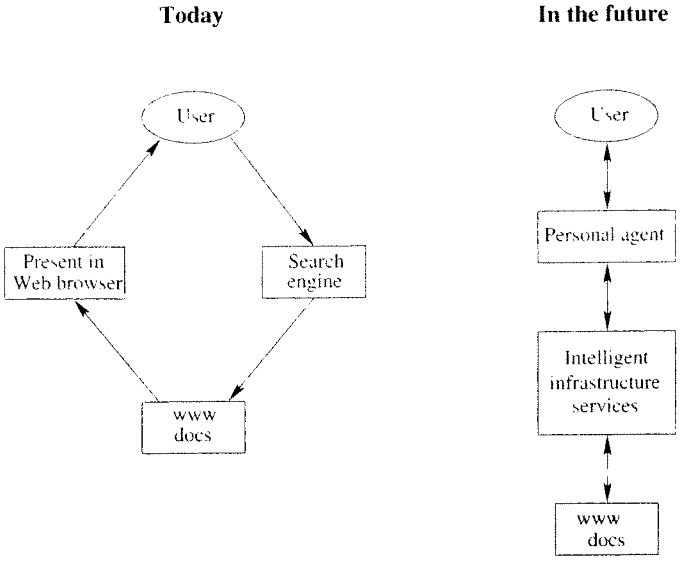

Figure 1.2 Intelligent personal agents

- Logic will be used for processing retrieved information and for drawing conclusions.

Further technologies will also be needed, such as agent communication languages. Also, for advanced applications it will be useful to represent formally the beliefs, desires, and intentions of agents, and to create and maintain user models. However, these points are somewhat orthogonal to the Semantic Web technologies. Therefore they are not discussed further in this book.

1.3.5 The Semantic Web versus Artificial Intelligence

As we have said, most of the technologies needed for the realization of the Semantic Web build upon work in the area of artificial intelligence. Given that AI has a long history, not always commercially successful, one might worry that, in the worst case, the Semantic Web will repeat AI's errors: big promises that raise too high expectations, which turn out not to be fulfilled (at least not in the promised time frame).

This worry is unjustified. The realization of the Semantic Web vision does not rely on human-level intelligence; in fact, as we have tried to explain, the challenges are approached in a different way. The full problem of AI is a deep scientific one, perhaps comparable to the central problems of physics (explain the physical world) or biology (explain the living world). So seen, the difficulties in achieving human-level Artificial Intelligence within ten or twenty years, as promised at some points in the past, should not have come as a surprise.

But on the Semantic Web partial solutions will work. Even if an intelligent agent is not able to come to all conclusions that a human user might draw, the agent will still contribute to a Web much superior to the current Web. This brings us to another difference. If the ultimate goal of AI is to build an intelligent agent exhibiting human-level intelligence (and higher), the goal of the Semantic Web is to assist human users in their day-to-day online activities.

Is Semantic Web $\supset$ AI ?

It is clear that the Semantic Web will make extensive use of current AI technology and that advances in that technology will lead to a better Semantic Web. But there is no need to wait until AI reaches a higher level of achievement; current AI technology is already sufficient to go a long way toward realizing the Semantic Web vision.

1.4 A Layered Approach

The development of the Semantic Web proceeds in steps, each step building a *layer* on top of another. The pragmatic justification for this approach is that it is easier to achieve consensus on small steps, whereas it is much harder to get everyone on board if too much is attempted. Usually there are several research groups moving in different directions; this competition of ideas is a major driving force for scientific progress. However, from an engineering perspective there is a need to standardize. So, if most researchers agree on certain issues and disagree on others, it makes sense to fix the points of agreement. This way, even if the more ambitious research efforts should fail, there will be at least partial positive outcomes.

Once a standard has been established, many more groups and companies will adopt it, instead of waiting to see which of the alternative research lines will be successful in the end. The nature of the Semantic Web is such that companies and single users must build tools, add content, and use that content. We cannot wait until the full Semantic Web vision materializes — it may take another ten years for it to be realized to its full extent (as envisioned today, of course).

In building one layer of the Semantic Web on top of another, two principles should be followed:

- Downward compatibility. Agents fully aware of a layer should also be able to interpret and use information written at lower levels. For example, agents aware of the semantics of OWL can take full advantage of information written in RDF and RDF Schema.

- Upward partial understanding. The design should be such that agents fully aware of a layer should be able to take at least partial advantage of information at higher levels. For example, an agent aware only of the RDF and RDF Schema semantics might interpret knowledge written in OWL partly, by disregarding those elements that go beyond RDF and RDF Schema. Of course, there is no requirement for all tools to provide this functionality; the point is that this option should be enabled.

While these ideas are theoretically appealing and have been used as guiding principles for the development of the Semantic Web, their realization in practice turned out to be difficult, and some compromises needed to be made. This will become clear in chapter 4, where the layering of RDF and OWL is discussed.

Figure 1.3 shows the "layer cake" of the Semantic Web (due to Tim Berners-Lee), which describes the main layers of the Semantic Web design and vision.

At the bottom we find *XML*, a language that lets one write structured Web documents with a user-defined vocabulary. XML is particularly suitable for sending documents across the Web.

RDF is a basic data model, like the entity-relationship model, for writing simple statements about Web objects (resources). The RDF data model does not rely on XML, but RDF has an XML-based syntax. Therefore, in figure 1.3, it is located on top of the XML layer.

RDF Schema provides modeling primitives for organizing Web objects into hierarchies. Key primitives are classes and properties, subclass and subproperty relationships, and domain and range restrictions. RDF Schema is based on RDF.

RDF Schema can be viewed as a primitive language for writing ontologies. But there is a need for more powerful *ontology languages* that expand RDF Schema and allow the representations of more complex relationships between Web objects. The *Logic* layer is used to enhance the ontology lan-

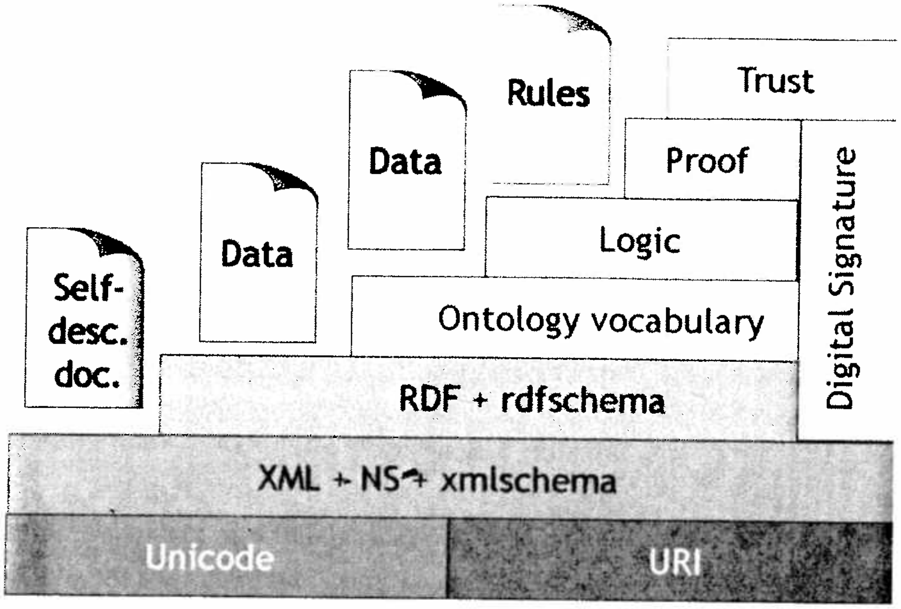

Figure 1.3 A layered approach to the Semantic Web

guage further and to allow the writing of application-specific declarative knowledge.

The *Proof layer* involves the actual deductive process as well as the representation of proofs in Web languages (from lower levels) and proof validation.

Finally, the *Trust layer* will emerge through the use of *digital signatures* and other kinds of knowledge, based on recommendations by trusted agents or on rating and certification agencies and consumer bodies. Sometimes "Web of Trust" is used to indicate that trust will be organized in the same distributed and chaotic way as the WWW itself. Being located at the top of the pyramid, trust is a high-level and crucial concept: the Web will only achieve its full potential when users have trust in its operations (security) and in the quality of information provided.

This classical layer stack is currently being debated. Figure 1.4 shows an alternative layer stack that takes recent developments into account. The main differences, compared to the stack in figure 1.3, are the following:

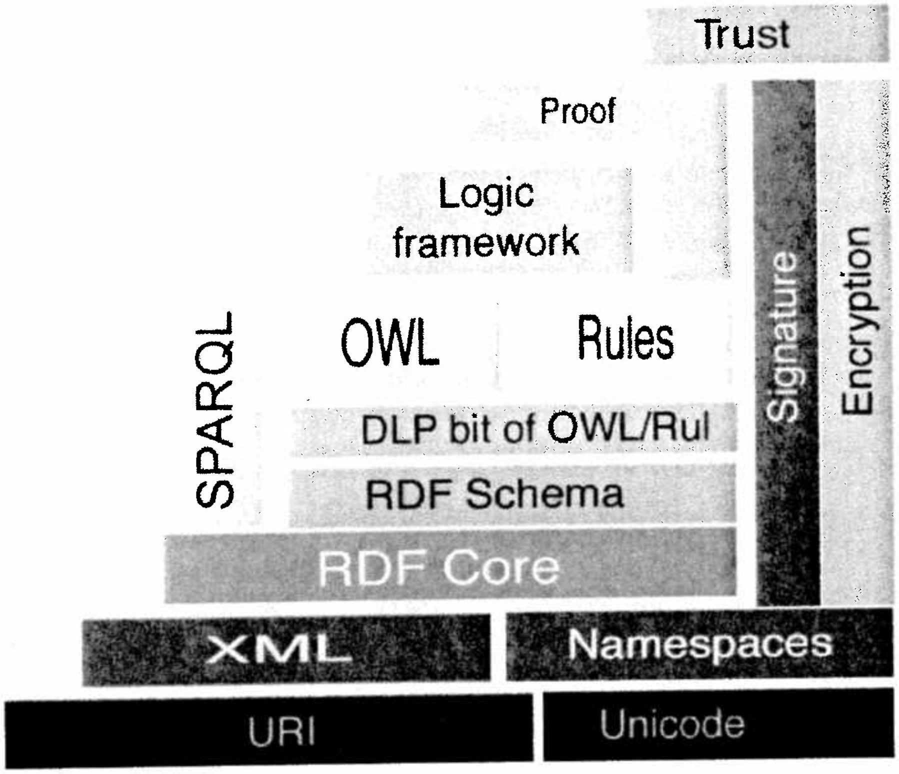

Figure 1.4 An alternative Semantic Web stack

- The ontology layer is instantiated with two alternatives: the current standard Web ontology language, OWL, and a rule-based language. Thus an alternative stream in the development of the Semantic Web appears.

- DLP is the intersection of OWL and Horn logic, and serves as a common foundation.

The Semantic Web architecture is currently being debated and may be subject to refinements and modifications in the future.

1.5 Book Overview

In this book we concentrate on the Semantic Web technologies that have reached a reasonable degree of maturity.

In chapter 2 we discuss XML and related technologies. XML introduces structure to Web documents, thus supporting syntactic interoperability. The structure of a document can be made machine-accessible through DTDs and XML Schema. We also discuss namespaces; accessing and querying XML documents using XPath; and transforming XML documents with XSLT.

In chapter 3 we discuss RDF and RDF Schema. RDF is a language in which we can express statements about objects (resources); it is a standard data model for machine-processable semantics. RDF Schema offers a number of modeling primitives for organizing RDF vocabularies in typed hierarchies.

Chapter 4 discusses OWL, the current proposal for a Web ontology language. It offers more modeling primitives compared to RDF Schema, and it has a clean, formal semantics.

Chapter 5 is devoted to rules, both monotonic and nonmonotonic, in the framework of the Semantic Web. While this layer has not yet been fully defined, the principles to be adopted are quite clear, so it makes sense to present them.

Chapter 6 discusses several application domains and explains the benefits that they will draw from the materialization of the Semantic Web vision.

Chapter 7 describes the development of ontology-based systems for the Web and contains a miniproject that employs much of the technology described in this book.

Finally, chapter 8 discusses briefly a few issues that are currently under debate in the Semantic Web community.

1.6 Summary

- The Semantic Web is an initiative that aims at improving the current state of the World Wide Web.

- The key idea is the use of machine-processable Web information.

- Key technologies include explicit metadata, ontologies, logic and inferencing, and intelligent agents.

- The development of the Semantic Web proceeds in layers.

Suggested Reading

An excellent introductory article, from which, among others, the scenario in section 1.2.5 was adapted.

- T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American* 284 (May 2001): 34–43.

An inspirational book about the history (and the future) of the Web is

- T. Berners-Lee, with M. Fischetti. *Weaving the Web*. San Francisco: Harper, 1999.

Many introductory articles on the Semantic Web are available online. Here we list a few:

- T. Berners-Lee. Semantic Web Road Map. September 1998. <http://www.w3.org/DesignIssues/Semantic.html>.
- T. Berners-Lee. Evolvability. March 1998. <http://www.w3.org/DesignIssues/Evolution.html>.
- T. Berners-Lee. What the Semantic Web Can Represent. September 1998. <http://www.w3.org/DesignIssues/RDFnot.html>.
- E. Dumbill. The Semantic Web: A Primer. November 1, 2000. <http://www.xml.com/pub/a/2000/11/01/semanticweb/>.
- F. van Harmelen and D. Fensel. Practical Knowledge Representation for the Web. 1999. <http://www.cs.vu.nl/~frankh/postscript/IJCAI99-III.html>.
- J. Hendler. Agents and the Semantic Web. *IEEE Intelligent Systems* 16 (March-April 2001): 30–37.
  Preprint at <http://www.cs.umd.edu/users/hendler/AgentWeb.html>.
- S. Palmer. The Semantic Web, Taking Form. <http://infomesh.net/2001/06/swform/>.
- S. Palmer. The Semantic Web: An Introduction. <http://infomesh.net/2001/swintro/>.
- A. Swartz. The Semantic Web in Breadth. <http://logicerror.com/semanticWeb-long>.

- A. Swartz and J. Hendler. The Semantic Web: A Network of Content for the Digital City. 2001. <http://blogspace.com/rdf/SwartzHendler.html>.

A collection of educational material related to the semantic web is maintained at REASE (Repository, European Association for Semantic Web Education):

- <http://rease.semanticweb.org/ubp>.

A number of Web sites maintain up-to-date information about the Semantic Web and related topics:

- <http://www.semanticweb.org/>.
- <http://www.w3.org/2001/sw/>.
- <http://www.ontology.org/>.

There is a good selection of research papers providing technical information on issues relating to the Semantic Web:

- D. Fensel, J. Hendler, H. Lieberman, and W. Wahlster, eds. *Spinning the Semantic Web.* Cambridge, Mass.: MIT Press, 2003.
- J. Davies, D. Fensel, and F. van Harmelen, eds. *Towards the Semantic Web: Ontology-Driven Knowledge Management.* New York: Wiley, 2002.
- The conference series of the *International Semantic Web Conference.* See <http://www.semanticweb.org/>.

Issues related to the Semantic Web architecture are discussed in the following article:

- I. Horrocks, B. Parsia, P. Patel-Schneider and J. Hendler. Semantic Web Architecture: Stack or Two Towers? In *Proceedings of the 3rd Workshop on Principles and Practice of Semantic Web Reasoning (PPSWR'05)*, LNCS 3703, Springer 2005, 37–41.

Information about semantic wikis is found at

- <http://wiki.ontoworld.org/wiki/Semantic_Wiki_Interest_Group>.