Part 1

Foundations and Technologies

1 Introduction to the Semantic Web Technologies

John Domingue[1] · *Dieter Fensel*[2] · *James A. Hendler*[3]
[1]The Open University, Milton Keynes, UK
[2]University of Innsbruck, Innsbruck, Austria
[3]Rensselaer Polytechnic Institute, Troy, NY, USA

John Domingue, Dieter Fensel & James A. Hendler (eds.), *Handbook of Semantic Web Technologies*,
DOI 10.1007/978-3-540-92913-0_1, © Springer-Verlag Berlin Heidelberg 2011

1.1 Introduction

- The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation [6].

For newcomers to the Semantic Web, the above definition taken from the article, which is often taken as the starting point for the research area, is as good a starting point as any. The goal of the Semantic Web is in some sense a counterpoint to the Web of 2001. That Web was designed as a global document repository with very easy routes to access, publish, and link documents, and Web documents were created to be accessed and read by humans.

The Semantic Web is a machine-readable Web. As implied above, a machine-readable Web facilitates human–computer cooperation. As appropriate and required, certain classes of tasks can be delegated to machines and therefore processed automatically. Of course, the design possibilities for a machine-readable Web are very large, and a number of design decisions were taken in developing the Semantic Web as it is seen today. The trade-offs in the design space are discussed later on in this chapter and also in the rest of the book. Two of the most significant are worth mentioning up front though. Firstly, as captured in the quote above, the Semantic Web is an extension of the Web. In particular, the Semantic Web builds upon the principles and technologies of the Web. It reuses the Web's global indexing and naming scheme, and Semantic Web documents can be accessed through standard Web browsers as well as through semantically aware applications. A global naming scheme means that in principle every semantic concept has a unique identifier, although in practice identity resolution is still a research area and the Semantic Web language OWL contains a specific relation to deal with this issue.

A second design choice is related to the fact that the Web is a shared resource, and therefore, within a machine-readable Web, meaning should be shared too. To this end, the Semantic Web incorporates the notion of an ontology, which by definition is a shared machine-readable representation (see ❯ Sect. 1.3.6). Through ontologies and ontology-related technologies, the meaning of and relationships between concepts within published Web pages can be processed and understood by software-based reasoners.

After about a decade of dedicated Semantic Web research, we are now entering a new phase for the technology. In short, it can now be claimed that the Semantic Web has arrived. There are a number of indicators to this. For example, semantic search engines now claim to index many millions of Semantic Web documents. Of course, this number of documents is small when compared to the size of the overall Web, but the trend resembles the early days of the Web, and if one counts the contained semantic statements (triples – see ❯ Sect. 1.3.4), then the number is estimated to be over a hundred billion triples.

Later in this chapter and also in most of the other chapters of this book, evidence is given to the take-up of Semantic Web technology. Semantics can be seen being deployed in a wide variety of settings including enterprise, government, media, and science arenas.

We are thus at a tipping point in the timeline of the Semantic Web where the technology can be seen to be moving out of research labs and into the mainstream in a nontrivial fashion.

To mark this juncture, this book describes the main technological components of the Semantic Web, the vertical areas in which the technology is being applied, and new trends in the medium and the long term. Each chapter covers general scientific and technical principles and also gives examples of application and pointers to relevant resources.

The rest of this chapter gives an introductory account of the notions of the Web and semantics from a technical perspective. Also, a brief history of the research area is discussed, given pointers to a number of general Semantic Web resources, and some highlights in terms of the deployment of semantic technology are outlined. The final section contains pointers to the future of the topic in general terms.

1.2 What Is the Web?

With over one trillion pages and billions of users, the Web is one of the most successful engineering artifacts ever created. At the end of 2009, there were 234 million websites of which 47 million were added in the year. The Web is now a rich media repository: the current upload to Flickr is equivalent to 30 billion new photos per year and YouTube now serves over one billion videos per day [50].

1.2.1 The Problem to Be Solved

As commonly known, the Web was invented by Sir Tim Berners-Lee while at CERN. The underlying problem he was tackling was how to manage and share technical information and knowledge at CERN where he was working at the time [5]. The overall scenario at the establishment contained several features, which can be found in many organizations over a certain size:

- The projects carried out were large and complex involving several different types of technologies.
- Work was carried by teams, which crossed CERN’s specified departments and unit structures.
- The knowledge involved was not static but rather changed over time.
- There was a rotation of staff. Workers came and went periodically – the typical length of stay at CERN at the time was 2 years.

This scenario led to the following underlying general requirements:

- Workers needed to be able to easily find and access relevant documents containing technical knowledge.
- The content of the documents needed to be easily changeable and the changes propagated across the organization quickly.

- The structure of the document collection could not be predetermined and had to be adapted easily.

The problems faced within CERN were acknowledged at the time to be relatively common and also ones that would become prevalent across the globe in the near to medium term as aptly expressed:

> ▸ CERN meets now some problems which the rest of the world will have to face soon [5].

1.2.2 Principles of the Web

As succinctly coined in the phrase: "For a hammer everything is a nail" (originally from [43, p. 15]), one has to be careful when differentiating between technological biases and the true underlying principles for any generic framework. Nevertheless, a significant portion of the design of the Web is based upon Hypertext, which was originally coined as a term by Ted Nelson [48] and has roots going back to Doug Engelbart's oNLine System [93] and Vannevar Bush's Memex system [11]. Another stream of innovation for the Web is based upon communication protocols, notably TCP-IP, a spin-off of TCP [12], which provides the bottom layer of the communication protocol for the Web.

Twenty years on from the starting points above, the principles of the Web are firmly established. These principles, many of which can be traced back to the original CERN proposal, have contributed significantly to the Web's success. These include:

- *Openness* Anyone or any organization can engage with the Web as a provider or consumer of information. Openness is an essential criterion for the success of the Web as a platform and incorporates:
  - *Accessibility* Web content can be accessed remotely from a wide variety of hardware and software platforms.
  - *Nonproprietary* The Web itself is not owned by any individual or organization, minimizing the effect cost has on participating.
  - *Consensual control* The Web structure is itself controlled and managed by an open body, the World Wide Web Consortium (W3C), which has a well-defined consensual process model for decision making.
  - *Usable* Usage of this infrastructure as a provider or user is kept as simple, smooth, and unrestricted as possible.
- *Interoperability* The Web is neutral to hardware and software platforms. A layer of protocols provides an integration mechanism, enabling heterogeneous proprietary and legacy solutions to interoperate through common interfaces.
- *Decentralized authorship and editorship* Content can appear, becoming modified, or be removed in a noncontrolled fashion. That is, the provisioning and modification of content is under the distributed control of the peers rather than being controlled by a central authority. Central control would hamper access and therefore scalability. A consequence of this principle is that an element of chaos or "untidiness" needs to be

tolerated. It is hard to imagine now, but in the early days of the Web one of the most common criticisms was that it would never take off because some Web pages could be found that were either incorrect or were below some quality threshold and also that some links were broken (two of the editors know of Computer Science professors who made this complaint).

- *Automated mechanisms are provided to route requests and responses* In order to scale, routing between requests and responses is handled in an automated fashion. Manual indexes or repositories are inherently nonscalable and costly, and immediately become outdated. The way that Web pages are accessed has changed over the past 10 years. At the beginning, one was required to know the IP-Address of the desired page and then later the URL (see below for a description). In this period, bookmark lists (especially lists of useful pages for a particular topic) were considered valuable intellectual property. Later, search engines such as AltaVista and Google raised access to the level of keywords.
- *Enabling n:m relationships* to maximize interaction. In contrast to email, where the content is targeted to specific receivers, the Web is based on anonymous distribution through publication. In principle, the information is disseminated to any potential reader, something that e-mail can only attempt to achieve through spam. The use of content for purposes not perceived by content producers facilitates serendipity on the Web and is one of the Web's key success enablers.

1.2.3 Web Architecture

The architecture of the Web is surprisingly simple for an engineering artifact with over a billion users. On the other hand, this is probably one of the main reasons for its success. From a functionality perspective, the Web provides the following:

- A *worldwide addressing schema*, which enables every document to have a unique globally addressable identifier. For the Web, this is provided by URLs (Uniform Resource Locators). A URL serves the purposes of both identifying a resource and also describing its network location so that it can be found. URIs (Uniform Resource Identifiers) encompass both URLs and URNs (Uniform Resource Names), where URNs denote the name of a resource.
- A *transport layer*, a protocol, HTTP (HyperText Transfer Protocol), which supports the remote access to content over a network layer (TCP-IP). HTTP functions as a request–response protocol in a client–server computing model. In HTTP, a Web browser typically acts as a client, while an application running on a computer host acts as a server.
- A *platform-independent interface*, which enables users to easily access any online resource. In case of the Web, it is HTML (HyperText Markup Language) and Web browsers that interpret and display the described content. HTML is thus a text and image formatting language, which is remotely served by Web host applications and used by Web browsers to display the Web content.

Integral to the makeup of the Web is the hyperlink which has its origins in the hypertext field. Hyperlinks allow a Web resource to point to any other Web resource by embedding the URL within an HTML construct (the "<a>" or anchor element). Links on the Web are unidirectional and are not verified, which means that links may break – the target Web resource may have been removed or the URL itself may be incorrect – leading to the "untidiness" mentioned earlier. However, not forcing links to be verified is widely accepted as being one of the design choices that enabled the Web to scale so quickly.

1.2.4 What Are the Problems with the Web?

The amount of information on the Web is staggering. The one trillion Web resources encompass practically every topic of human interest: from the life cycle of earthworms in New Zealand [110], to UK Pop Hits in the 1950s [66], to the Constitution of Mauritius [44].

Accessing documents can be efficient on the Web; if one knows the right keywords then extremely so – to the point where experienced users would rather search for the PDF of a paper online than get up out of their chairs and access a hardcopy on the shelf. The usefulness of document search can be seen from the fact that in December 2009 it was noted that 87.8 billion searches were conducted each month on Google [61]. As an extension to the Web, the Semantic Web has been created to solve two specific problems, which are as follows:

- *Accessing data* – the "standard Web" is limited in that:
  - *Documents are indexed and accessed via plain text*, that is, a string-based matching algorithm is used to retrieve documents according to a given request. This creates problems for ambiguous terms, for example, "Paris" can denote: the capital of France; towns in Canada, Kiribati, and the USA; a number of films including "Paris, Texas" by Wim Wenders; fictional characters including the legendary figure from the Trojan War; and a number of celebrities including the daughter of Michael Jackson, and Paris Hilton the socialite and heiress. Moreover, complex matching involving inference is not feasible without additional technology. For example, correctly answering the query: "where can I go on holiday next week for 10 days with two young children for less than 1000 Euros in total?" is not possible with current search engines.
  - The current paradigm is dominated by *returning single "best fit" documents for a search*. Often, the answer to a query is available on the Web but requires the combination and integration of the content of multiple source documents. The dominant search engines today leave this integration of content to the user.
  - *Underlying data are not available*. A significant number of websites are generated through databases but the underlying data are hidden behind the presented HTML. This phenomenon is sometimes termed "the dark Web" and significantly hinders the usability and reusability of the underlying information. A way to overcome this problem is to "Web scrape" the data by parsing the presented

HTML. This process though is error-prone and unstable with regard to changes in the way the page is displayed (e.g., if the layout or color scheme is altered). It should be noted that the concept of making legacy database data available was specified as a requirement in the original proposal from Sir Tim Berners-Lee.

- *Enabling delegation* – the Web can be viewed as a very large collection of static documents. When users browse the Web, their computers act simply as rendering devices displaying text and graphics and sometimes audio and video content. All inference and computation is left to the user. To a large extent, the computational abilities of the computational device are not used. Coupled with the above ones on users to carry out their own inferences, the sheer volume and growth of data available creates a strong need for at least some level of automation. For example, current estimates are that the 281 exabytes (10^6 TB) of information created or replicated worldwide in 2007 will grow tenfold by 2011 to 1 zettabyte (10^9 TB) per year. Delegating tasks such as the integration of information, data analysis, and sense-making to machines, at least partially, is the only way forward for users, communities, and businesses to continue to make the most of the information available on the Web.

Given the above requirements, the Semantic Web extends the Web with "meaning" supporting access to data at web-scale and enabling the delegation of certain classes of tasks. As the Web has documents at the center, the Semantic Web places data and the semantics of data at its core. An overview of the architecture of the Semantic Web is given in ❯ Semantic Web Architecture.

1.3 What Are Semantics?

Computer science, since the early beginning, has been concerned with processing of data. Programming languages provide simple and complex datatypes to store data. Originally, the semantics of these data were hardwired in the programs in which they were interpreted and used. Around 50 years ago, data began to become separated from the application program to be stored in **databases**. This allowed one to reuse the same data in different programming contexts and prevented the same data management component being re-implemented across many applications. The fact that the meaning of the data was no longer hardwired directly into the application program led to mechanisms for representing the structure and semantics of the data being developed. One such extremely successful structure was the relational data model (cf. [23]). In addition to simple data that can be aligned easily with the constructs of programming languages, a growing number of documents in natural language started to be placed within computers in the 1960s. Unfortunately, relational database technology is not a very useful or efficient paradigm to store, manipulate, and query these types of documents. In consequence, the areas of **information retrieval** (cf. [41]), **information extraction** (cf. [46]), and **natural language processing** (cf. [34]) evolved in parallel. These areas are concerned with capturing the meaning contained in digital natural language documents to support their automatic processing. A third area of computational semantics was founded around 1955 with the goal of enabling a computer to act intelligently as humans do, that is, generating **Artificial Intelligences** (cf. [54]). The field began by implementing general problem solving methods such as global search and theorem proving. However, after a short space of time, the numerical complexity of the tasks involved in intelligent problem solving made it apparent that a machine-understandable representation of the knowledge related to how a problem may be solved efficiently was required.

▸ Knowledge of the specific task domain in which the program is to do its problem solving was more important as a source of power for competent problem solving than the reasoning method employed [17].

The subareas of **knowledge representation** (cf. [8]) and **knowledge acquisition**, which was later called **knowledge engineering** (cf. [55]), were created to provide methods and techniques to represent human knowledge in a machine-understandable manner.

All these areas of Computer Science focus on capturing the meaning of data in a machine-processable manner and provide the historical context from which semantic technology was developed. The following briefly discusses the essential essence of semantic technology, as well as its form and substance.

1.3.1 Semantics, the Science of (Meaning)^2

Semantic technology provides machine-understandable (or better machine-processable) descriptions of data, programs, and infrastructure, enabling computers to reflect on these artifacts. Now, what does machine-processable semantics really mean? Let us ask Wikipedia, the world leading resource of human knowledge. Let us specifically ask for **machine-processable semantics**. Unfortunately, there is no direct response. Okay let us ask for its three elements.

A *machine* is any *device* that uses *energy* to perform some activity [92]. Okay, one now needs to understand what a *device* is. Here, get a pointer to Wiktionary: "Any piece of *equipment* made for a particular *purpose*" [104]. By the way, only equipment that uses energy qualifies as a machine. Still, what is equipment and why does it require a *purpose*? Let us ask Wikipedia again. *Equipment* redirects to *tools*. Okay, let us check *tools*. "A *tool*, broadly defined, is an entity that interfaces between two or more domains;.... Basic tools are *simple machines*" [88]. Basic machines are somehow simple machines? Well, yes, but...? The aspect of *energy consumption* has still not been explored that distinguishes a machine from a generic *device*, and *purpose* that distinguishes a *device* from the more generic *equipment*.

- "In all such *energy transformation* processes, the total energy remains the same" [87]. What was meant by *consuming* energy? "*Energy* is a *quantity* that can be assigned to every *particle*" [87]. Here, proceedings become a bit philosophical. Trying to find out what a *quantity* is and why it is that it can be assigned to all *particles* will be resisted. Not to mention that the notion of an *assignment* should really be investigated and delved into whether *particles* or *waves* are the final truth? It does not really help to distinguish between a *machine* and a *device*. That is, *machines* remain defined as being *machines* (more precisely, it is learnt only that basic machines are simple machines).

- "*Purpose* is a *result*, end, aim, or *goal* of an action *intentionally* undertaken" [95]. So what is an *intention*? "An *agent's intention* in performing an *action* is his or her specific *purpose*" [91]. No, there will be no attempt to find out what an *agent* is.

Processable does not have a hit at Wikipedia. This saves both time and space.

"*Semantics* is the study of meaning, . . . This problem of understanding has been the subject of many formal inquiries. . . most notably in the field of *formal semantics*" [98]. Also from the same source: "The word '*semantics*' itself denotes a range of *ideas*." Fortunately only the *word*. And no, we will not try to understand what an *idea* is, since already in the narrowest sense "an *idea* is just whatever is before the *mind* when one thinks" [90]. Let us try to find out the meaning of formal semantics: "*Formal semantics* is the study of the semantics" [89]. Okay, formal semantics is the study of semantics and semantics is the study of meaning. Obviously meaning is the study of? **No**, meaning "is the end, purpose, or significance of something" [64]. So, formal semantics is the study of the study of purpose. Purpose is to remember the attribute used to distinguish a device from generic equipment (which is a machine if it consumes energy).

Naively entered here is an infinite regression of circular definitions written in natural language. This would be an opportune moment to refer to the importance of cooperation as a grounding mechanism for communication and to conduct a detailed analysis of the role of vocal and nonvocal communication mechanisms (cf. [45, 56]) in order to escape this infinite regress. However, this is not the focus here. Obviously, life is a circle and one needs to be pragmatic. Let us try to understand the essence of semantic technology through its usage starting with a number of predecessor technologies.

What is the main value of a traditional relational *database*? According to Wikipedia, "a *database* is a collection of data" and "the term *data* means groups of information" . . . "*Information* as a concept has many meanings . . ." The authors do not tell us whether information that is not viewed as a concept would have less meaning. According to Wikipedia, *meaning* also has many meanings. Still, Oracle is able to successfully sell *bases of collections of groups of information that have many meanings when viewed as a concept not mentioning the fact that already meaning has many meanings*. Moreover, Oracle makes billions of dollars per annum with this kind of rather vague business.

In a relational database, everything is represented in a table, and a row has a key and a column has a name. With this, even with a very simple machine, one can find the phone number of Mr. `X` if `X` is the value of the `name` column and `phone number` is the heading of another column. Unfortunately, with an average Web page, this is far more difficult. As mentioned earlier, hidden in various HTML tags there is a name (a random alphanumerical string similar to many others) and somewhere else a phone number (a set of integers including some special characters). A browser is required to render the information and a human reader to understand the information based on the layout of the website. This is the solution as implemented in the Web which was introduced 20 years ago. As outlined earlier, the sheer simplicity has made the Web an incredible success story with now more than one billion users. Its simplicity also leaves room for improvement.

Semantic technology adds tags to semistructured information as database technology adds column headings to tabular information. Let us use a small example:

```
<person>
  <name>Sir Tim</name>
  <phone number>01-444444</phone number>
</person>
```

These annotations allow a computer "to understand" that *Sir Tim* is a name of a person and *01-444444* is his telephone number. In a similar fashion, programs and other computational resources can be described through semantic annotations. *This is the essence of Semantic Web technology.*

What can be seen from this example is that one needs two things to define the semantics of information: a language such as <X>Y</X> to define the meaning of Y, and terms such as X to denote this meaning. This is investigated in more detail in the following.

1.3.2 Form

Logic is a 1,000-year-old technology to formally capture meaning. Over this long history, especially relatively recently, a large number of logics have been developed, each suitable for a specific purpose. The focus is on a small number of these languages, in particular, on those that provide insights into the overall design issues associated with logical languages and those that have been applied in a Semantic Web context. A number of languages will be then examined that are used to express the meaning of data on the Semantic Web. Finally, there would be a discussion on open issues and problems when applying logic to the Web.

1.3.3 Logic

From an algorithmic perspective, implementing logical-reasoning systems demonstrates clearly how complex decidability and complexity are to manage (cf. [29, 35]). First, briefly described are logical paradigms in increasing levels of complexity, and then, how computer scientists identified reasonable subsets which can be handled to a certain extent.

Propositional Logic is a rather simple logic language providing propositions such as A, B, C,... and logic connectives such as AND and OR. All interpretations are simply the enumerations of all possible false and true assignments to these propositions. Therefore, propositional logic is decidable, although, already NP-hard.

First-Order Predicate Logic provides a richer means to define such propositions by providing terms such as c, f(c, X),... and predicate symbols that can be applied to these terms P(c), Q(c, f(c,X),... . Terms can make use of variables that can be existentially or all quantified (i.e., either there must exist a term fulfilling a formula or all terms must fulfill a formula). First-order predicate logic is still semi-decidable. That means, there are complete and correct evaluation methods; however, it is not possible to guarantee that they terminate. An important feature of first-order logic is the distinction between terms and predicates, that is, one is not allowed to apply predicates or terms to predicates.

Second-Order Predicate Logic [96] and comparable languages drop this limitation (cf. [13]). Here, one can apply predicates to other predicates or entire formulae and interpret variables as sets rather than as individuals of a domain of interpretation. Unfortunately, for these languages, already unification, that is, the question of whether two terms can be substituted, is semi-decidable, which means that there is not even an approach for implementing inference in these languages. The question of how far one can make progress in simulating second-order features syntactically (statements over statements or classes that can be instances of other classes) in a semantic first-order framework has been explored in F-Logic (cf. [37]) and more generally in HiLog (cf. [13]).)

In layman terms, propositional logic is reasoning about individuals. It is decidable but the effort grows exponentially with the number of individuals. First-order logic is reasoning over sets of individuals (each predicate is interpreted as a set), which is complete but does not guarantee a terminating decision procedure. Second-order logic is concerned with reasoning about sets that have elements which may again be sets. The focus of **computational logic** is on identifying subsets of logic that can be handled by computers. Unfortunately, what one gets here is not necessarily what one would need.

Most approaches in automatic theorem proving and software verification use variants of **first-order logic** to reason (cf. [53]). Here, based on the transformation of the general clause form, resolution and unification (cf. [8]) provide a complete although only semi-decidable decision procedure. Obviously, for this level of expressiveness, only incomplete reasoning requiring heuristic guidance can be achieved in the general case.

A restriction of the pragmatic complexity can be achieved by restricting first-order logic to **Horn logic** and applying Selective Linear Definite resolution [99]. There are also variants that forbid or cleverly restrict the usage of function symbols creating a decidable language – propositional logic with some additional syntactical sugar. Most work on Horn logic alters the model theory of logic by not considering all models but models that are defined through certain *minimality* criteria (this model is unique in the case where negation in the bodies of the Horn clauses is either restricted or does not exist, cf. [39]). In layman terms, this model assumes that only facts which can be inferred are true and that all other facts are false. This is called the *closed-world assumption* and originates from the database area. A well-known implementation of this paradigm is Prolog (cf. [14]). Interestingly enough, this paradigm extends the expressiveness of these syntactically restricted first-order languages beyond first-order logic as it becomes possible to express the transitive closure of a relationship.

Description Logics (cf. [3]) provide a whole family of sub-languages of first-order logic of differing complexity. Common among these languages is to restrict the formalism to *unary and binary predicates* (concepts and properties) and to restrict the usage of function symbols and logical connectors to build complex formulae. The different levels of complexity and the decidability of these languages follow from the precise definition of these restrictions. Therefore, many different languages have been defined and implemented, many of which contain intractable worst-case behavior but which however still work for many practical applications (cf. [30]).

1.3.4 Semantic Web Languages

HTML provides a number of ways to express the semantics of data. An obvious one is the `META` tag [108]:

```
<META name = "Author" lang= "fr" content = "Arnaud Le Hors">
```

In the time before the wider usage of RDF, systems such as Ontobroker (cf. [19]) used the attribute of the anchor tag to encode semantic information (see the › Sect. 1.5). It is also possible to interpret the semantics of HTML documents indirectly. For example, information captured in a heading tag of level one (`<H1>`) may be used to encode concepts that are significantly important for describing the content of a document. Still, HTML was not designed to provide descriptions of documents beyond that of informing the browser on how to render the contents. Within efforts to stretch the use of HTML to include meaning, the term semantic HTML was created – see [97] for more details on this.

The **Extensible Markup Language (XML)** [109] has been developed as a generic way to structure documents on the Web. It generalizes HTML by allowing user-defined tags. This flexibility of XML, however, reduces the possibilities for the type of semantic interpretation that was possible with the predefined tags of HTML.

The **Resource Description Framework (RDF)** (cf. [42]) is a simple data model for semantically describing resources on the Web. Binary properties interlink terms forming a directed graph. These terms as well as the properties are described using URIs. Since a property can be a URI, it can again be used as a term interlinked to another property. That is, unlike most logical languages or databases, it is not possible to distinguish the language or schema from statements in the language or schema. For example, in the statement `<rdf:type, rdf:type, rdf:Property>` it is stated that `type` is of `type property`. Also, unlike conventional hypertext, in RDF, URIs can refer to any identifiable thing (e.g., a person, vehicle, business, or event). This very flexible data model is obviously suitable in the context of a free and open Web; however, it generates quite a headache for logicians who wish to layer a language on top. More details on RDF can be found in [107] and in › Semantic Annotation and Retrieval: RDF.

RDF schema (RDFS) (cf. [9]) uses basic RDF statements and defines a simple ontology language. Specifically, it defines entities such as `rdfs:class`, `rdfs:subclass`, `rdfs:subproperty`, `rdfs:domain`, and `rdfs:range`, enabling one to model classes, properties with domain and range restrictions, and hierarchies of classes and properties. RDFS is a specific RDF vocabulary for this purpose and is simply RDF plus some more definitions (statements) in RDF.

The **Web Ontology Language OWL** (cf. [16]) extends this vocabulary to a full-fledged spectrum of Descriptions Logics defined in RDF, namely, OWL Lite, OWL DL, and OWL Full. Mechanisms are provided to define properties to be inverse, transitive, symmetric, or functional. Properties can be used to define the membership of instances for classes or hierarchies of classes and of properties. Frankly, OWL Lite is already quite an expressive Description Logic which makes the development of efficient implementations for large data sets quite challenging and, in practice, as difficult as implementing OWL DL. However, neither of these languages can make use of full RDF, that is, some valid RDF statements are not valid in Lite or DL. This is due to the fact that logic languages such as Descriptions Logics exclude meta statements, that is, statements over statements. For RDF and RDFS, this was not a problem since neither language provided mechanisms to define complex logical definitions. Spoken in a nutshell, Lite and DL define a vocabulary in RDF **and** restrict the usage of RDF. OWL Full drops these restrictions. OWL Full provides the vocabulary of OWL DL, that is, an expressive Description Logic, and allows for any valid RDF statement. For example, in OWL Full, a class can be treated simultaneously as a set of individuals and as an individual. Therefore, OWL Full is beyond the expressive scope of Description Logic and minimally requires a theorem prover type of inference such as first-order logic (i.e., is semi-decidable).

Still, OWL Full can be used as a basis to find useful restrictions (OWL DL is an example of such a restriction) and generate useful languages such as the **Simple Knowledge Organization System (SKOS)** (cf. [33]). SKOS is a data model for knowledge organization systems that uses keywords to describe resources. SKOS is defined as an OWL Full ontology, that is, it uses a sub-vocabulary of OWL Full to define a vocabulary for simple resource descriptions based on controlled structured vocabularies.

OWL2 (cf. [47]) started in 2007 to address some of the issues around OWL. In particular, OWL Lite had been defined as an overexpressive Description Logic. This hampered the implementation of Lite reasoning based on existing semantic repository technologies and also made the layering of rules on top of the language unfeasible. Specifically, there was too big a gap between RDFS and OWL Lite. In consequence, three new sub-languages were defined. OWL2EL provides polynomial time algorithms for all the standard reasoning tasks of description logic, OWL2QL enables efficient query answering over large instance populations, and **OWL2RL** restricts the expressiveness with respect to extensibility toward rule languages. OWL2RL seamlessly links with rule-based presentations of RDFS and extensions to simple rule languages (cf. [32], [52]). This is currently the route that most industrial semantic repository developers follow and will probably define together with OWL2QL the most important Semantic Web representation languages from a technological point of view.

The **Rule Interchange Format (RIF)** (cf. [36]) complements OWL with a language framework centered on the rule paradigm. Like OWL, it does not come as a single language but as a number of sub-languages. The framework incorporates RIF-BLD, which defines a simple logic-oriented rule language; RIF-PRD, which captures most of the aspects of production rule systems; and RIF-Core, which is the intersection of both these languages. This split is due to the fact that the W3C working group had to cover two very different paradigms which are only similar at the surface level: rules based on a declarative interpretation of logic (cf. [39]) and rules that model event–action systems based on the production rule paradigm (cf. [24]). The former usually have a declarative semantics in terms of a variation of a minimal Herbrand model and were an alternative model for databases called deductive databases. The latter normally only have an operational semantics and are used to express the dynamic aspects of processes. Production rules are in essence a kind of programming language based on a blackboard architecture and event triggers. Since these production systems are no longer called expert system shells but business rule engines (suitable to implement business processes), they have gained significant commercial interest. Creating a merger of these two different paradigms was a nontrivial task. Finally, these three dialects are complemented by The Framework for Logic dialects (RIF-FLD) as a way to define new RIF dialects. RIF uses XML as the exchange syntax and unfortunately does not directly layer on top of RDF.

Since RDF is a data model, it also requires a query language. As SQL [100] is a means to express queries over relation databases, **SPARQL** (cf. [51]) is a query language for the graph-based data model of RDF. SPARQL has been developed without considering RDFS, OWL, and RIF (see *Fig. 1.2*). More details on the query language can be found in Querying the Semantic Web: SPARQL.

Up to now formats to create metadata statements have been discussed, but not how to link these to existing Web content. Returning to the earlier example:

```
<person>
  <name>Sir Tim</name>
  <phone number>01-444444</phone number>
</person>
```

A way to define a concept `person` and properties such as `name` and `phone number` has been developed, but there is yet no mechanism to express that `Sir Tim` is the name of a person. Grounding or connecting metadata with documents on the Web is supported by a set of languages. **Microformats** [75] are predefined formats to add meta information to elements in HTML and XML. A well-known microformat is `hCard,` which can be used for representing people, companies, organizations, and places, using `vCard`, a file format standard for electronic business cards. These formats not only provide a language structure to present information but additionally provide domain-specific terminologies (controlled vocabularies) for this purpose. Therefore, they directly interweave structure and content. **RDFa** (cf. [1]) provides a set of XHTML attributes to include RDF metadata directly into HTML and XML documents. In contrast to Microformats, RDFa does not predefine domain-specific terminologies. **GRDDL** has been developed as a mechanism for **G**leaning **R**esource **D**escriptions from **D**ialects of **L**anguages to derive RDFa definitions from Microformats (cf. [28]) helping to integrate information from heterogeneous sources. More information on Microformats, RDFa, and GRDDL can be found in Semantic Annotation and Retrieval: Web of Hypertext – RDFa and Microformats. Documents are, however, only one type of data source available on the Web. In addition to being a global repository for human-readable documents, the Web is becoming more and more a platform for applications and application integration. Within the Web of Data (cf. [7] and also Semantic Annotation and Retrieval: Web of Data), billions of semantically described data items have been made available for applications to consume and process. The majority of data is generated from relational databases and there has been recent associated W3C effort, **R2RML,** to define mappings from relational models to RDF, that is, connecting databases with semantic metadata. As stated in [106], "The **mission** of the RDB2RDF Working Group is to standardize a language for mapping relational data and relational database schemas into RDF and OWL, tentatively called the RDB2RDF Mapping Language, R2RML." Finally, one can consider the Web from a perspective of services that provide functionality either for other services or human users. Attaching semantics to services can be achieved through **Semantic Annotations for WSDL and XML Schema** (**SAWSDL**) (cf. [38] and also ➲ Semantic Web Services).

1.3.5 The Tower of Babel

- The Open Systems Interconnection model (OSI model) is a product of the Open Systems Interconnection effort at the International Organization for Standardization. It is a way of sub-dividing a communications system into smaller parts called layers. A layer is a collection of conceptually similar functions that provide services to the layer above it and receives services from the layer below [94].

This model is widely used in designing network architectures on a global scale. A model starts with the physical layer and ends with the application layer that provides mechanisms such as the HTTP protocol. For example, in the Internet stack, the Internet protocol components IP and TCP are at levels 3 and 4. Sir Tim Berners-Lee started a similar conceptual effort to structure the Semantic Web (see ➲ *Fig. 1.1*).

At the lowest level, Unicode is seen as a means to encode text, URIs to refer to resources, and XML with its namespace and schema mechanisms to provide syntactic descriptions of structured objects. On top of this, he envisioned five layers of semantics: RDF, OWL, RIF, and layers for proof and trust. This type of layering has two major functions: preventing an

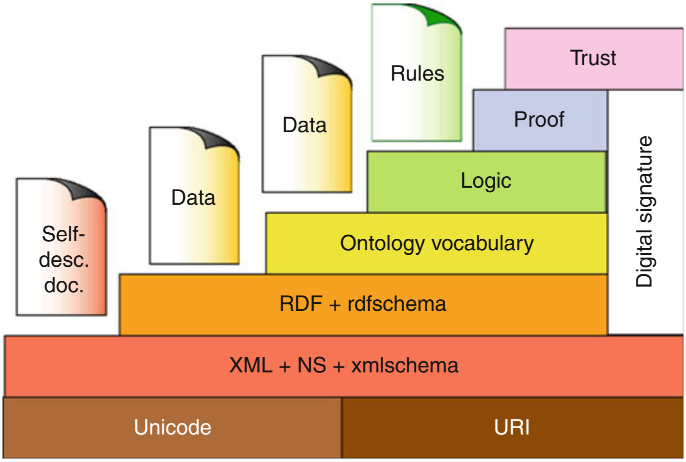


Fig. 1.1
The Semantic Web Layer Cake – 1

upper layer from re-implementing functionality provided by a layer below and allowing an application that only understands a lower layer to at least interpret portions of definitions at a higher layer.

- 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 [2].

For example, OWL should not define a new `owl:Class` statement but rather reuse the already provided `rdfs:Class` statement.

Ideally, an RDFS-aware agent may not understand a property restriction for an OWL class but at least it would understand some of the elements of a class definition in OWL. Unfortunately, this is not the case.

- The rationale for having a separate OWL class construct lies in the restrictions on OWL DL (and thus also on OWL Lite), which imply that not all RDFS classes are legal OWL DL classes [105].

That is, OWL does **not** layer properly on top of RDF and RDFS (cf. [49]). This also breaks the second compatibility of [2]:

- 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.

Unfortunately, this is also **not** the case! Even worse, these faults in layering OWL on top of RDF properly are not due to the fact that our colleagues involved in the language were incompetent. It actually reflects a fundamental problem associated with layering logic on top of the RDF. As outlined before:

- RDF allows arbitrary statements over statements and reflects an intrinsic property of the Web.
- OWL Lite and OWL DL as first-order logic pedantically distinguish statements in the language from statements about the language which are kept strictly separated.

Obviously, this creates conflict and only experience can show how this fundamental problem can be resolved in a pragmatic manner that best fits practical needs. Note that statements over statements (and, e.g., statements over logic connectives such as AND and OR inside the language) is even beyond second-order logic and requires self-referenciability with all its paradox conclusions, such as allowing to express an RDF statement that states that it is not an RDF statement. A radical outcome could be that logic is not well suited for the Semantic Web, however; what else could play this role?

Another issue of layering is internal to OWL. As previously mentioned, OWL Lite was too powerful a Description Logic to be really distinguishable from OWL DL in computational terms. Here, a central design concern of OIL Light (cf. [20]) was ignored. The possibility to establish a coherent extension to RDFS that also enabled the possibility to layer rules on top was missed. Meanwhile, with the less expressive sub-language profiles in OWL2, this has now been repaired, and obviously OWL Lite will be less than a footnote in the development of the Semantic Web. ➔ KR and Reasoning on the Semantic Web: OWL contains a comprehensive overview of OWL.

An early layering proposal for a rule language on the Web was SWRL (cf. [31]). SWRL neatly layered a rule language on top of OWL, that is, as an extension of the already available OWL vocabulary. Unfortunately, this layering did not capture the essence of either Description Logics or of rule languages. Both are defined as fragments of first-order logic to reduce the computational complexity of executing inference. When simply combining them, this feature gets lost. As a result, one has a syntactic restriction of first-order logic without any gain in computational terms. Only when one restricts the rules to DL-safe rules is decidability restored. Simply, OWL Lite was too powerful a Description Logic to be used as a starting point for a feasible rule language. This problem is actually reflected in an update of the layer cake, as presented in ➔ *Fig. 1.2*. You may notice in this figure that *proof* is no longer a proper layer, that a query language is developed as an alternative to the logic stack, and finally that there is a wish for the Holy Grail, a unifying logic.

In conclusion, RIF was developed in parallel to OWL. Actually, it views XML as an exchange syntax and, as mentioned previously, is not defined as a layer on top of RDF (see ➔ KR and Reasoning on the Semantic Web: RIF). It is therefore somewhat isolated from the other languages associated with the Semantic Web.

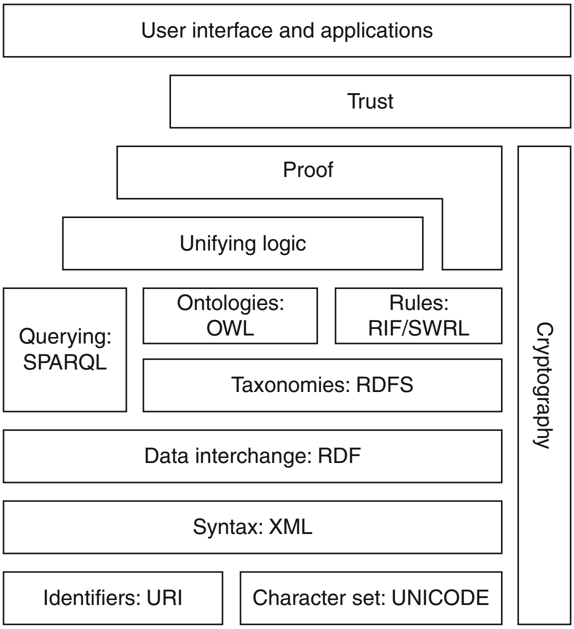


Fig. 1.2
The Semantic Web Layer Cake – 2

As already mentioned earlier, most rule languages slightly alter the semantics of first-order logic by not using all possible models but a specific (minimal) model. This comes along with what is called the **closed-world assumption**. If a fact is not evaluated to be true in this model, it is assumed to be false. This goes beyond the expressive power of first-order logic (which OWL is based on). Here, simply a truth value will not be assigned to it, since it is not restricted to a specific model. That is, it is not inferred that a fact is false from the situation where a fact is not known to be true in a specific model. This is termed the **open-world assumption**. As the Web is an open world, an open-world assumption sounds like a suitable proposition. However, with the same rationale, one could also argue for reasoning based on the closed-world assumption in relation to the portion of the Web one is investigating. This difference between rule and Description Logic languages is also reflected in the way they interpret integrity constraints, such as the domain and range restrictions of properties. When the value of a property is found and it is not known that it is a member of its range, it is assumed that there must be a mistake. The violation of a constraint is indicated over the range of the property. This is how most rule languages work. It is **not** known that a fact holds and one therefore assumes its **negation**. OWL does the opposite. OWL would infer that this value must be an element of the set defining the range of the property since the integrity constraint is requesting this. Frankly, it is hard to tell which type of reasoning is most suitable for the Web. Therefore, the designer of RDFS took a wise decision:

- For example, an RDF vocabulary might describe limitations on the types of values that are appropriate for some property, or on the classes to which it makes sense to ascribe such properties. The RDF Vocabulary Description language provides a mechanism for describing this information, but does not say whether or how an application should use it. For example, while an RDF vocabulary can assert that an `author` property is used to indicate resources that are instances of the class `Person`, it does not say whether or how an application should act in processing that range information. Different applications will use this information in different ways. For example, data checking tools might use this to help discover errors in some data set, an interactive editor might suggest appropriate values, and a reasoning application might use it to infer additional information from instance data. RDF vocabularies can describe relationships [9].

Already from this statement, you can trace the branching of OWL and RIF.

RIF has the fundamental problem of covering rule languages based on very different paradigms incorporating either a declarative or an operational flavor. It is of no surprise that RIF is not a single language but, within its first version, provides three languages. OWL now provides at least six different dialects. Thus in total, one has more than ten Semantic Web languages, and RIF additionally contains a framework for defining more. This language fragmentation is quite dangerous as it may significantly hamper information interoperability between Semantic Web applications and also significantly increase the effort to implement them.

As a final example, let us examine the layering of SKOS. First of all, SKOS uses RDF and OWL. Therefore, it should be assumed that SKOS is layered on top of OWL. However, it is simpler than OWL. OWL is supposed to provide a language for defining ontologies, and SKOS is a way to define simple "taxonomies." Therefore, it does not extend OWL but rather defines a small extension of a heavily constrained restriction of OWL. In general, one would naively expect to define OWL as an extension of SKOS. Not to mention that SKOS is agnostic in regard to whether its restricted version of OWL is interpreted as OWL DL or OWL Full. In the end, the Semantic Web is closer to the tower of Babel than to a coherently layered network protocol stack, and Yahweh, the enemy of global communication, may succeed again [103]. Moreover, currently there is no theoretical technique that one can apply to select one of these languages as the "right" language. Maybe the wisdom of the crowd or swarm intelligence may solve this issue in terms of impact. One may also worry a little less given the fact that holy logic also has a similar problem in that rule languages syntactically restrict and semantically extend first-order logic. What a layering!

1.3.6 Substance

For defining machine-processable metadata, a formal language for definitional purposes is required and also for linking to content available on the Web. In addition, terms are needed to actually write down metadata statements.

The simplest technique is to support keyword lists taken from a natural language. This is often called a **tag**. These tags can be freely chosen or predefined by a **controlled vocabulary** (this type of tagging is also called "subject indexing") [86]. Folksonomies as used at Web 2.0 websites are an example of the former. Users can freely define tags, and tag clouds indicate the most popular term for a subject [101]. In library science, controlled vocabularies are widely used. However, it is not enough to simply control the vocabulary; one must also control its usage. There are various studies indicating that people will choose different terms to annotate a resource and that these terms may not be necessarily useful when a user is searching for the resource and is not familiar with the vocabulary. A controlled vocabulary can be based on a **thesaurus** such as WordNet [85] that groups nouns, verbs, adjectives, and adverbs into sets of synonyms (i.e., concepts).

The next step is to use a **taxonomy** [102], which is a classification schema arranged in a hierarchical structure. Simple taxonomies can be formalized in RDFS that provides hierarchies of classes and properties. When adding formal definitions to state that a certain value of a property must be fulfilled in order to classify it as an instance of a certain class, one can use language elements of OWL or RIF. **Ontologies** (cf. [22]) are discussed in detail in ❯ *Ontologies and the Semantic Web*, and so would be discussed lightly here. A very common definition of ontologies attributed to Gruber [25] is that *an ontology is a formal, explicit specification of a shared conceptualization.* Each of these attributes denotes the following:

- **Formal** – the specification is represented in a formal language which is machine processable. For the Semantic Web, this means one of the standard representation languages such as RDF or OWL.

- **Explicit** – as appropriate underlying assumptions are written down. There is a design trade-off as to how much of a domain should be contained in a specification: the level of granularity (how fine-grained) and the level of abstraction or genericity. The dimensions of this design space include:
  - *Usability* – this includes both being understandable by developers or a targeted community and also the match between the conceptualization and the requirements associated with the tasks and software applications that the ontology is used within.
  - *Reusability* – minimizing dependence with any specific task, software component, or other ontology.
- **Specification** – an ontology is a description of the artifact and is independent from the entity described. This is most meaningful when the target domain covers IT resources such as software components.
- **Shared** – an ontology only makes sense if it is shared by a community of use. The purpose and benefit of ontologies in a Semantic Web context is that they support interoperability between the designer or producer of a resource and the (software-underpinned) user. A set of formal statements hidden on a single machine does not fulfill the definition nor the purpose of an ontology.
- **Conceptualization** – an abstract simplified view of a domain of interest which is required for some task or purpose. Following from this, one thus expects ontologies to have a level of coherence and completeness with respect to a certain domain.

Note that one views all the metadata formats discussed earlier as ontologies which vary in the level of formalization. Examples of widely used ontologies are Dublin Core [65] for describing resources through properties such as title, creator, subject, publisher, etc., and **Friend Of A Friend (FOAF)** [68] that defines a set of properties such as name, e-mail address, home page, and interests to describe and link people.

1.4 Semantics and the Web

Over the last 10 years, there have been a number of ways in which different communities have envisioned how semantics and the Web can be combined. Each of these has in part been due to the research areas from which the communities originally came from and partly related to a particular conceptualization of what a semantically enhanced Web would look like. It is worth reflecting on these in order to appreciate the Semantic Web as a research topic.

1.4.1 The Semantic Web as a Layer over Text

A number of the issues raised here are covered in ❯ *Semantic Annotations and Retrieval: Manual, Semiautomatic, and Automatic Generation* technically in depth. However, it is still worth exploring some of the underlying issues. Even 10 years ago, when the Semantic Web began to take off, the Web was large (7 million unique sites [78]). Moreover, more so than now it could be characterized as a large collection of text. From the beginning of the Semantic Web as a research project, there was a view that the key problem was how to connect with the current Web as a text resource, that is, how to transform a Web of millions or billions of text documents into a well-structured and well-defined repository of semantically described assets.

Relatively quickly a number of issues emerged. Text on the Web is not the same as text found in non-Web documents (e.g., company reports) which previous Natural Language Processing (NLP) research had focused on. Specifically, on the Web:

- Text can be shorter, comprising short phrases or single words.
- Text can be ungrammatical.
- The interpretation of text can rely on the underlying HTML-based structure, for example, laying out multiple columns in a table or the font used.

Because of the above, most successful NLP approaches to the Semantic Web rely on no or only shallow parsing.

As well as the input to the systems being different, differences in the required output also led to a stream of research. Information Extraction (IE) technologies, able to identify known entities, such as people, places, and organizations, had initial successes when applied to the Web, but these systems tended to produce unconnected entities, for example, that "John Lennon" is a person and "Imagine" is a song, missing out the relation between the two.

A more general issue associated with the above is the generic way in which NLP and ontology-based-reasoning components were integrated in applications. For the most part, these components were placed as black boxes, which were pipelined together. Only recently, in projects such as LarKC [74], has significant effort been put into combining algorithms associated with the two research areas.

A final issue related to the Semantic Web and NLP has been how to relate the (newly produced) semantic data to the original text. Trade-offs in this design space included:

- Minimizing the additional data added to the original Web page
- Facilitating the reuse of the data accumulated
- Supporting maintenance when the original Web page is altered

As mentioned above, some of the issues described here are outlined in ❯ Semantic Annotations and Retrieval: Manual, Semiautomatic, and Automatic Generation.

1.4.2 Semantic Web as a Database

The Semantic Web as a research area saw the coming together of a number of communities including Artificial Intelligence (from agents, knowledge modeling, and logic) and the Web. For the most part, though, the research overlap between the Semantic Web and databases was minimal. This could be seen as somewhat surprising as the Web of Data is now a widely used term, but, in the early days, the emphasis was on creating knowledge structures as a platform for agents (see below).

The emergence of linked data as described in detail in ➲ *Semantic Annotation and Retrieval: Web of Data* and the use of linked data in initiatives such as those described in ➲ *eGovernment* have given rise to a stream of research which brings together the Semantic Web and database communities. RDF stores are now seen from the academic and industrial sectors, which can be deployed in settings where performance is a key issue. For example, below is outlined how an RDF triple store was used to support the BBC Sport's pages during the 2010 World Cup, which received millions of page requests per day [57].

Commercial successes such as mentioned above have now led to a more detailed discussion with the overall goal of bringing the logic and data close together. The main research issues that are currently beginning to emerge include the following:

- Which particular database techniques (e.g., partitioned hashes, column tables) are most applicable to high-performance RDF storage?
- How to structure benchmarks for large-scale repositories? Including what are the correct dimensions?
- When and where should reasoning be handled? For example, materialization (the precomputation and storage of inferred triples) is an expensive process which may not contribute to desired results.

These issues are discussed in detail in ➲ Storing the Semantic Web: Repositories. Another contribution to this debate is the Billion Triple Challenge run in conjunction with the International Semantic Web Conference (see below in Related Resources) [60]. Finally, Orri Erling has an interesting database-centric blog on this in [76].

1.4.3 Semantic Web as a Platform for Agents

From the beginning, the Semantic Web was seen as a necessary platform for supporting agents which could carry out tasks on behalf of human users. Within the seminal Semantic Web paper [6], a scenario is presented at the start where a Semantic Web agent books an important medical appointment checking the online diaries of a woman, her two grown-up children, and a number of hospitals satisfying geographic and quality constraints. The motivation for creating the Semantic Web is based on the functionality provided by software agents, which rely on the combination and exchange of content from diverse sources. The Semantic Web would allow agents to read the content of pages because the data are coded in a machine-readable representation. The underlying ontological basis for the data supports semantic interoperability by coding meaning in a way that supports semantic mediation.

Given the early motivations, however, the amount of agent research based on Semantic Web technology has been relatively small. There were two main reasons for this. Firstly, more emphasis than initially envisioned was required for creating a robust, usable, and scalable data layer. Also the majority of agent research was founded on FIPA protocols [67] rather than the stack of Web standards. Reevaluating the Semantic Web agent vision in light of newer phenomena such as the Web of Data and the Social Web (see ➲ Social Semantic Web) would be an interesting research exercise.

Research in Semantic Web Services, covered in ➲ Semantic Web Services, has also been seen as a means to provide an infrastructure for Semantic Web agents, but this has not been widely pursued.

1.5 Brief History

It is hard to know who first had the idea of creating a language on the World Wide Web that could be used to express the domain knowledge needed to improve Web applications. By the mid-1990s, before most people even knew the Web existed, several research groups were playing with the idea that if Web markup (which was all primarily HTML) contained some machine-readable "hints" to the computer, then one could do a better job of Web tasks like search, query, and faceted browsing. It is important to note that at that time, the potential power of the Web was still being debated, and there were many who were sure it would fail (see the ➲ Sect. 1.2.2).

However, by 1997 or so, it was clear that the Web was going to be around for some time, and there was a burst of energy going on. Various researchers were publishing algorithms, suggesting that different approaches could be used for searching the Web rather than the traditional AI approaches, and it was around this time that Sergey Brin and Larry Page published their famous "PageRank" paper [10], which led to the creation of Google and the growth of the modern search engine. This historical event is mentioned here as it is sometimes said that the Semantic Web was created to improve search. This is partly true, but it is important to note that search as known back then, pre-Google, was not the same as the current keyword search that powers so much of the modern Web today.

At this time, the first "real" refereed publications were also seen coming about machine-readable knowledge on the Web. One of these approaches was the SHOE (Simple HTML Ontology Extensions) project, which took place at the University of Maryland [40]. The slogan for the SHOE project, which continues to be a popular quote in the Semantic Web community, was "*A little semantics goes a long way*," and supporting this slogan the SHOE Base Ontology contained a very minimal set of concepts and relationships. Around this effort, a number of tools were created within the project including a semantic annotator for HTML pages and a semantic search tool.

Another early project was Ontobroker [18], which, like SHOE, looked at adding and using semantic annotations to HTML pages. These two early projects looked at what is now called Web ontology languages, and were driven less by the AI-inspired push for expressive languages, and more by the needs of the emerging Web – what would now be called semantic annotation or tagging.

Other early projects within Europe included On-To-knowledge (which began in January 2000) from which the SESAME repository was developed and also OIL [20] was set up as a cross-project initiative, merging an effort called XOL (XML Ontology Language) and the work emerging from Ontobroker. Approximately 18 months later the OntoWeb [70] network of excellence started, which was the birthplace for the Knowledge Web project [71].

In parallel with this Web representation work, W3C had begun to explore whether some sort of Web markup language could be defined to help bring data to the Web. The Metadata C Format working group was drafting a language that was later to be named the Resource Description Format (RDF). There was at this time a split between XML and RDF, which we do not have space here to recount but suffice to say that this added confusion to the overall story.

It is also worth noting here the dialogue that began in the late 1990s within the Knowledge Acquisition Workshop Series in Banff [72] on the relationship between knowledge acquisition, modeling, and the Web. One of the projects that came out of this discussion was IBROW^3 [4], which examined how knowledge components could be reused through the Web. Elements of this project later influenced Semantic Web Services research (see ➲ Semantic Web Services).

1.5.1 Increasing Research Interest

In 1999, one of the editors began a 3-year position as a funding agent for the US Defense Advanced Research Projects Agency and convinced them to invest in the technology. The primary argument was that Semantic Web technology could be used to help solve a lot of the Department of Defense's (and, of course, everyone else's) data integration problems. To help sell the US government on funding this research area, the techniques pioneered in Ontobroker and SHOE were used to build some demos showing the potential for these new languages.

Based on these demos, a project called the DARPA Agent Markup Language (DAML) was launched. MIT's Semantic Web Advanced Development, led by Sir Tim Berners-Lee, was funded under this program, with a proposal to base a language on top of RDF which was at the time being defined. RDF, like SHOE, used URIs to name concepts, an important aspect of "webizing" the representation languages for the Web. Along the way, the community (both research and industrial) came to accept Tim's name for this work: The Semantic Web.

In actuality, it is worth noting that the Semantic Web was a realization of part of Tim's original conception of the Web. In fact, in a 1994 talk (Web Conference, Geneva) he said:

- Documents on the web describe real objects and imaginary concepts, and give particular relationships between them... For example, a document might describe a person. The title document to a house describes a house and also the ownership relation with a person.... This means that machines, as well as people operating on the web of information, can do real things. For example, a program could search for a house and negotiate transfer of ownership of the house to a new owner. The land registry guarantees that the title actually represents reality.

As this work grew, it was decided that an effort was needed to bring together the key players in this emerging area. The outcome of this was a Dagstuhl Seminar held in 2000 [62]. The workshop was quite successful and led to a dramatic increase in funding especially in Europe. For example, ➲ *Fig. 1.3* below shows the snapshot of the projects funded by the

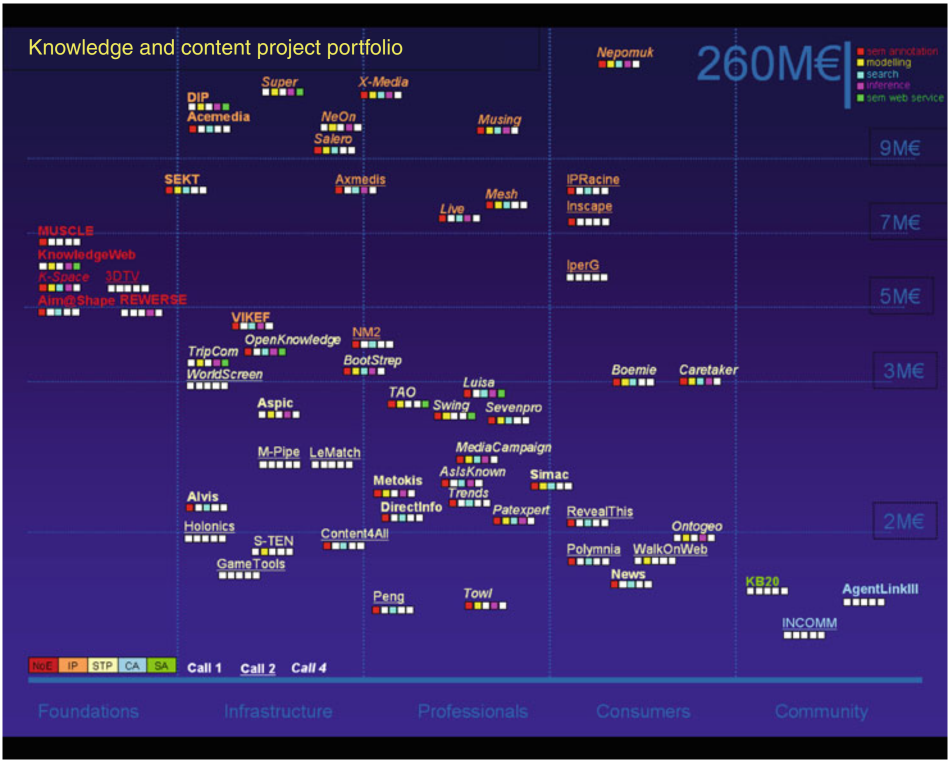


Fig. 1.3

A snapshot of the projects funded by the Knowledge and Content Unit in Luxembourg in 2005. This slide is available within a set at ftp://ftp.cordis.europa.eu/pub/ist/docs/kct/iswc05-slideshow_en.pdf (Figure used with the permission of the European Commission)

Knowledge and Content Unit in Luxembourg by type and funding, color coded according to the areas of semantic annotation, modeling, search, inference, and Semantic Web Services.

1.6 Related Resources

1.6.1 Semantic Web Events

1.6.1.1 Conferences

- **Asian Semantic Web Conference** (ASWC) – a Semantic Web conference series that targets the Asian continent. See http://www.sti2.org/conferenceseries/asian-semantic-web-conferences for details on the overall conference series.
- **European Semantic Technology Conference** (ESTC) – this conference tackles the commercial aspects for semantic technology with a European focus and is usually held

in Vienna, Austria. See http://www.sti2.org/conferenceseries/european-semantic-technology-conferences for details on the overall conference series.

- **Extended Semantic Web Conference (ESWC – formerly the European Semantic Web Conference)** – this annual conference had its seventh edition in 2010 and includes workshops and tutorials. The change in name relates to the conference series covering topics related to the application of semantics to mobile platforms, cloud computing, sensor networks, as well as the Web. See http://www.sti2.org/conferenceseries/extended-semantic-web-conferences for details on the overall conference series.
- **International Semantic Web Conference (ISWC)** – this annual conference is now (2010) in its ninth year and is a premier event for discussing Semantic Web topics. The event usually attracts around 600 participants and includes a research and in-use track as well as workshops and tutorials. See http://iswc.semanticweb.org/ for details on the overall conference series.
- **I-Semantics –** is a European forum that examines semantics from a technological, economic, and social point of view. Details on the conference series can be found at http://i-semantics.tugraz.at/.
- **SemTech** – is an annual event that targets the professional area and the commercial deployment of semantic technology. This conference is usually held in San Francisco in the USA. Details on the 2010 event can be found at http://semtech2010.semanticuniverse.com/.
- **IEEE International Conference on Semantic Computing (ICSC)** – addresses the use of computational semantics to create, use, manage, and find content, where content refers to any type of resource including video, audio, text, processes, services, hardware, and networks. More details can be found at http://www.ieee-icsc.org/.
- **World Wide Web Conference** – this conference provides a forum for debate and discussion on the evolution of the Web, the standardization of the associated technologies, and the impact of the technologies on society and culture. This conference traditionally includes a Semantic Web track. More details can be found at http://www.iw3c2.org/.

1.6.1.2 Summer Schools and Tutorials

- **ESWC Summer School** – is a new Semantic Web summer school that will be held in conjunction with the ESWC conference. Details on the 2011 event can be found at http://summerschool.eswc2011.org/.
- **IEEE Summer School on Semantic Computing** – is a week-long event that up until now has been held on the Berkeley campus in California. See http://www.sssc2010.org/ for details on the 2010 event.
- **Introduction to Semantic Web Tutorial** – has been held as a one-day event in conjunction with ISWC 2007, 2008, and 2010. See http://people.csail.mit.edu/pcm/SemWebTutorial.html for details on the 2010 event.

- **Summer School on Ontological Engineering and the Semantic Web** – this week-long summer school, which started in 2003, was initially funded by the EU OntoWeb and later the KnowledgeWeb project, and has always been held in Cercedilla, near Madrid, Spain. Details on the 2008 summer school can be found at http://kmi.open.ac.uk/events/sssw08/.

1.6.1.3 Semantic Web Journals and Magazines

- **Journal of Web Semantics** – this journal covers the main areas associated with the Semantic Web and publishes research, survey, ontology, and systems papers. More details on the journal can be found at http://www.elsevier.com/wps/find/journaldescription.cws_home/671322/description#description.
- **IEEE Intelligent Systems** – is a magazine that covers the broad area related to systems that act intelligently. It often includes papers though related to the Semantic Web. More details can be found at http://www.computer.org/portal/web/intelligent/home.
- **Applied Ontology** – covers conceptual modeling and ontology analysis. Details on the journal can be found at http://www.iospress.nl/loadtop/load.php?isbn = 15705838.
- **Semantic Web: Interoperability, Usability, Applicability** – is a Semantic Web journal that uses an open and transparent review process. Submitted manuscripts are posted on the journal's website to which researchers are free to post public reviews and authors to post responses. More details on the journal can be found at http://www.semantic-web-journal.net/.
- **International Journal On Semantic Web and Information Systems** – is a journal where aspects of the Semantic Web relevant to the Computer Science and Information Systems communities are discussed. See http://www.ijswis.org/ for more details.

1.6.1.4 Semantic Websites

- http://www.iswsa.org/ – the Web page for the Semantic Web Science Association that runs ISWC
- http://semanticweb.org/ – a Wiki page for the Semantic Web community
- http://www.sti2.org/ – contains a list of resources including events that are organized by STI International, a networked organization for parties interested in Semantic Technology
- http://www.w3.org/2001/sw/ – the W3C page that lists W3C Semantic Web activities
- http://www.linkeddata.org – provides a home for and pointers to resources associated with the linked data initiative
- http://data.semanticweb.org/ – the Semantic Web Conference Corpus also known as the Semantic Web Dog Food Corpus, which contains data and ontologies related to Semantic Web events (including ESWC, ISWC, and WWW mentioned above) and researchers, organizations, and papers related to the area

1.6.1.5 Sources Introducing the Semantic Web

A number of videos and websites exist that outline the basic notions behind the Semantic Web.

- http://videolectures.net/iswc08_hendler_ittsw/ – the Introduction to the Semantic Web Tutorial from ISWC 2008
- http://www.youtube.com/watch?v = OGg8A2zfWKg – a very clear introduction to the Semantic Web from Digital Bazaar Inc.
- http://infomesh.net/2001/swintro/#whatIsSw – a simple and comprehensive introduction for anyone trying to understand the Semantic Web

1.6.1.6 Books

- **A Semantic Web Primer** (2nd Edition) Grigoris Antoniou and Frank van Harmelen (MIT Press) – a textbook suitable for undergraduates which gives a broad introduction to the motivation behind the Semantic Web, as well as its applications and supporting technologies. The book introduces the specific languages associated with the Semantic Web including RDF and OWL. Additional material including slides can be found at http://www.semanticwebprimer.org/.
- **Foundations of Semantic Web Technologies** by Pascal Hitzler, Markus Krötzsch, and Sebastian Rudolph (Chapman and Hall) – this book covers RDF Schema, OWL, rules, and query languages, such as SPARQL. Recent developments such as OWL 2 and RIF are also covered.
- **Semantic Web for Dummies** by Jeffrey T. Pollock (Wiley Inc.) – provides a gentle introduction to the Semantic Web covering the area as a set of technologies, a social phenomena, and a web-scale architecture.
- **The Semantic Web: Semantics for Data and Services on the Web** by Vipul Kashyap, Christoph Bussler, and Matthew Moran (Springer) – covers the Semantic Web from a data and process perspective and includes basic coverage of XML, RDF, and ontologies.
- **Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL** by Dean Allemang and James Hendler (Morgan Kaufmann) – is a practical book aimed at practitioners who wish to create semantic models using Semantic Web technologies.

1.7 Selected Successes in the Commercial Sphere

During the just over decade covering the Semantic Web as a research topic, one of the most common criticisms was that the work would never be commercially successful due to problems with the scalability and usability of semantic technology. The debate on whether Semantic Web technologies will be commercially successful is now over and has been replaced instead with a discussion on what specific forms deployed commercial semantic applications will take. Moreover, a number of commercial announcements have been made recently, which indicate that one is moving from an early adopters phase to more mainstream markets for semantic technologies. A longer discussion of this can be found in ❯ *Semantic Technology Adoption: A Business Perspective*.

1.7.1 Oracle

Oracle's support for semantic technology started with its 10gR2 system and a number of enhancements were made when 11g was subsequently released. On their website [77], Oracle now state that 11g supports a number of core technologies including RDF(S), OWL, SKOS, and SPARQL. Also, support is provided for a number of open-source tools, including Jena, Sesame, and Protégé, and a number of third-party entity extraction services, such as OpenCalais and GATE.

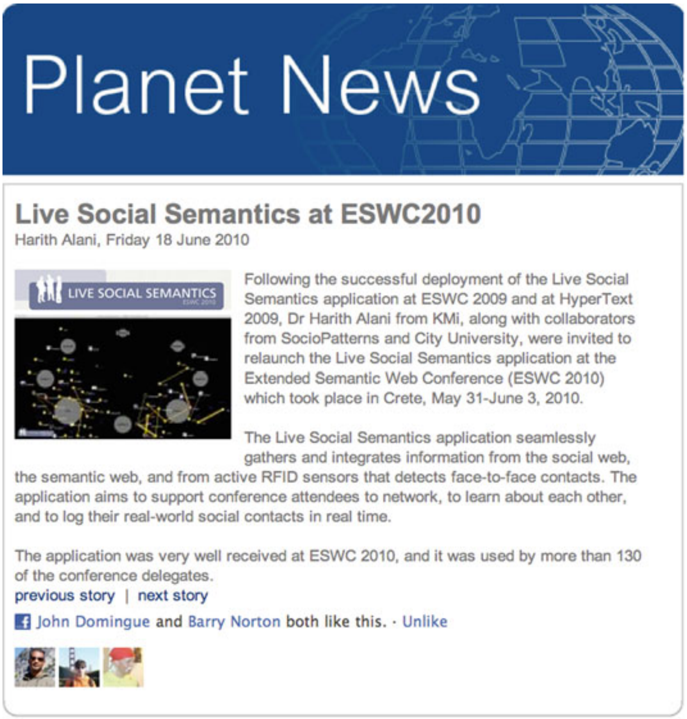
A technical perspective on 11g including benchmarking information can be found in ❯ Storing the Semantic Web: Repositories. However, of interest here is the fact that a mainstream conventional IT provider is now an advocate of the Semantic Web. One of the main reasons for this is that as with many commercial shifts, this was a requirement from Oracle customers, particularly in the areas of pharmaceutics, life sciences, and health care, who need to integrate large amounts of data from many different sources. This type of data integration at scale and across many heterogeneous sources which cannot be changed is one where semantic repositories cope well. Additionally, in these areas, reasoning capabilities are useful in supporting the mining and analysis of the data.

1.7.2 Facebook's Open Graph Protocol

In May 2010, Facebook announced their Open Graph Protocol [63], which is based on RDFa. The exact relationship between Open Graph and RDFa is discussed in ❯ *Semantic Annotation and Retrieval: Web of Hypertext – RDFa and Microformats* in detail. Here, the focus is on the impact of the announcement. In short, the Open Graph protocol facilitates the integration of Web resources into a Facebook social graph. A Facebook "like" button can be embedded in any Web page allowing Facebook users to "like" any Web resource. ❯ *Figure 1.4* below shows this facility in use in an Open University news system enabling readers to express preferences over published stories. It is seen in the figure that three readers have expressed that they like the story. These preferences also allow site owners to track the demographic data of users visiting their site.

In the last few months, a number of commercial companies have built sites around this feature. Levi's have a dedicated store, which incorporates a like button for every product [79]. Also, Amazon have integrated their recommendation system to use Facebook profiles through Open Graph. Facebook have also recently integrated Open Graph into the Facebook SDK for the iPhone and Android platforms.

There are two main reasons for highlighting this deployment of semantic technology. Firstly, now in effect there are 500 million (and currently growing) Facebook users



Live Social Semantics at ESWC2010

Harith Alani, Friday 18 June 2010

Following the successful deployment of the Live Social Semantics application at ESWC 2009 and at HyperText 2009, Dr Harith Alani from KMi, along with collaborators from SocioPatterns and City University, were invited to relaunch the Live Social Semantics application at the Extended Semantic Web Conference (ESWC 2010) which took place in Crete, May 31-June 3, 2010.

The Live Social Semantics application seamlessly gathers and integrates information from the social web, the semantic web, and from active RFID sensors that detects face-to-face contacts. The application aims to support conference attendees to network, to learn about each other, and to log their real-world social contacts in real time.

The application was very well received at ESWC 2010, and it was used by more than 130 of the conference delegates.

previous story | next story

John Domingue and Barry Norton both like this. · Unlike

Fig. 1.4

A screen snapshot from the online news system of the Knowledge Media Institute where Facebook users can say that they like a story

semantically annotating the Web from fixed and mobile devices. The probability is that this will in the short to medium term be a major source for semantic data. When making the announcement, Facebook's CEO, Mark Zuckerberg, claimed that the technology would result in over one billion like buttons spreading across the Web in the first 24 h [81].

The second more general aspect about the announcement is that one of the world's largest Web companies deems Semantic Web technology a suitable choice on which to center its corporate strategy. In particular, Facebook currently claim that Open Graph is "the most transformative thing we've ever done for the web" [83] – which is a very strong endorsement for semantic technology.

1.7.3 Google Buys Metaweb

In July 2010, Google bought Metaweb, the company which maintains Freebase. As reported in Semantic Annotation and Retrieval: Web of Data, Freebase is a major source of cross-domain data within the growing linked dataset. Currently, Freebase has around 12 million items including movies, books, and organizations. According to Google's

Director of Product Management, Freebase will enable the company to target more complex questions such as "actors over 40 who have won at least one Oscar?" [69]. From a linked data viewpoint, one interesting aspect of this purchase is that Google intends to maintain Freebase as a free and open resource. This announcement builds upon Google's use of microformats and RDFa to power their Rich Snippets feature, which is used to enhance returned search results.

1.7.4 BBC Football World Cup 2010 Website

For the 2010 Football World Cup, the BBC website used a semantic-based publishing framework based on an RDF triple store described in Storing the Semantic Web: Repositories. The Website included over 700 pages describing the 32 teams, 8 groups, and the associated hundreds of footballers that took part in the event. The Web pages were dynamically aggregated using a football ontology describing concepts associated with the World Cup (e.g., teams, players, and groups) as well as publication assets (e.g., story, blog, image, and video).

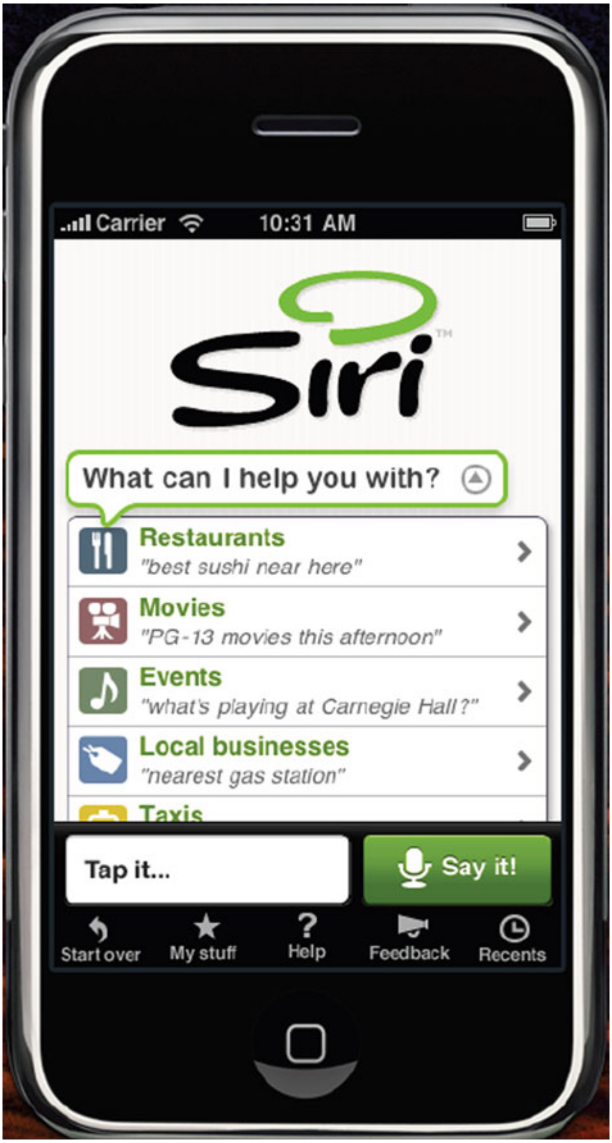
One can see the page describing the England midfielder Frank Lampard. Using the underlying ontology and the stored RDF data, the page shows the basic statistics for Frank's performance in the World Cup: the number of games played, the number of goals scored and goal assists, the number of shots on and off target for the goal, and also statistics related to discipline, such as yellow and red cards, and the number of fouls committed by and committed on Frank. The key advantage here of course is that the page is generated dynamically from the data and, thus, the publication process is streamlined and maintenance effort is drastically reduced (see http://news.bbc.co.uk/sport1/hi/football/world_cup_2010/groups_and_teams/team/england/frank_lampard).

The use of semantic technology was deemed to be successful and the website proved popular dealing with several million page requests every day throughout the World Cup. BBC now plans to use the technology again for the London Olympics in 2012 and the Chief Technical Architect, Journalism and Knowledge, BBC Future Media and Technology stated: "We look forward to seeing the use of Linked Data grow as we move towards a more Semantic Web" [58].

Technical details on the above can be found in Storing the Semantic Web: Repositories.

1.7.5 Apple Buys SIRI

Siri is a free iPhone App, currently only available in the USA, which acts as a virtual personal assistant for a set of common tasks. *Figure 1.5* shows the main interface for Siri. User requests, which can be typed or spoken, are given through a dialog interface customized for smart phone screens. Context information, including the user's location and personal preferences, the time, and the selected task, are used to aid in understanding



What can I help you with?

Restaurants
"best sushi near here"

Movies
"PG-13 movies this afternoon"

Events
"what's playing at Carnegie Hall?"

Local businesses
"nearest gas station"

Taxis

Fig. 1.5

A screen snapshot of the SIRI interface [27] showing the interface for the iPhone (Image courtesy: Tom Gruber, © Siri)

the posed request. The currently supported tasks include booking a table at a restaurant, for a movie, or for an event, and requesting a local taxi or finding local businesses.

In *Fig. 1.6*, the role that semantics plays within the overall architecture can be seen. In addition to the sophisticated dialog system, domain and task models are used to support the combining of online services to fulfill the requested task. There is in fact

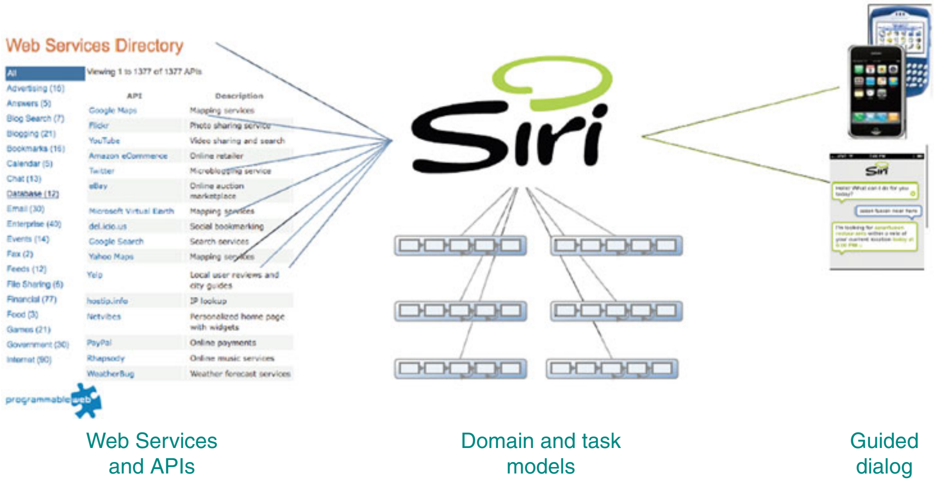


Fig. 1.6

The Siri overall architecture [26] where domain and task models are used to combine online services and Web APIs to satisfy user requests given from a mobile device (Image courtesy: Tom Gruber, © Siri)

a (partly historical) commonality between the approach that Siri takes to combining services and the WSMO [21] approach highlighted in Semantic Web Services. One of the main functionalities provided by semantics in the Siri architecture is in providing the mapping between a task "book a table for 2 people at a Mexican restaurant in the local vicinity" and online services (restaurant finding services, recommendation services, restaurant table booking services, etc.).

The main benefit that Siri provides for the end user is that a simple conversation replaces the effort of combining either a sequence of Web searches or a sequence of mobile phone App interactions. Siri had raised approximately $24 million in venture funding and was bought by Apple in the summer of 2010 for an estimated value of between $100 and $200 million [59].

It can be seen above that semantic technology is beginning to enter the mainstream. Also, by and large, it is the simpler technologies which are data-centric that have been taken up. There are a number of views that one could take on this. One is that it should be expected that by their very nature, real-world Web applications will be dominated by data rather than conceptual structures. Second, even with the successes emerging now, the Semantic Web is still in a preliminary phase of commercialization and it will take time to progress to Web applications, which require more complex conceptual reasoning.

The acquisition of Siri runs somewhat counter to the reasoning above and indicates that there may be space for more complex forms of reasoning, as is required to deal with services and Web APIs.

1.8 Future

Chapter on ❯ Future Trends contains predictions of semantic technologies 5, 10, and 15 years into the future from application and core technology points of view. Reflecting on the last decade of research into the Semantic Web, two issues seem clear. Firstly, as outlined above, at this point semantic technology is becoming mainstream and we will continue to see deployment of semantics in the commercial sector. It is envisaged that in the near term, organizations will make significant portions of their data available on the Web using semantic technologies. Moreover, the emergence of data will grow in a way analogous to the way in which the Web grew. At the beginning of the Web, it was often asked what would motivate individuals and organizations to put resources into creating and developing websites. Over the history of the Web, we have seen a progressive escalation in this effort. Corporations will now have entire departments dedicated to maintaining their presence on the Web. Web presence is seen as a requirement rather than a luxury, and the Google ranking of an organization can determine its success. As a first step toward the vision outlined in the *Scientific American* paper [6], a semantic data presence will soon become a requirement rather than a luxury. When advocating that semantic technology would be a core pillar of the UK's Digital Britain initiative, Gordon Brown (when he was the UK Prime Minister) declared one significant benefit would be the reduction in the cost of maintaining government websites [73]. Thus, linked data moves the effort of creating and maintaining websites and Web applications over organizational data to external parties. Chapter ❯ Knowledge Management in Large Organizations discusses related issues from an enterprise perspective.

Secondly, the Web is changing in a number of ways. As covered in ❯ *Social Semantic Web* and mentioned briefly above, there is already a link between social networking sites and the Semantic Web. It is expected to see a growth in platforms for Web applications based upon combinations of social networking and semantic technologies, harnessing the power of human networks and automated reasoning. A discussion is currently taking place related to which forces will dominate the way the Internet is used. Wired recently ran an article with the title "The Web Is Dead. Long Live the Internet" [84]. In this article, the authors saw three trends emerging. Firstly, that video and peer-to-peer network traffic are beginning to take a large proportion of Internet traffic when compared to pure Web communication. Secondly, that as predicted in several places, the number of users accessing the Internet from mobile devices will soon surpass the number who access it from PCs. A consequence of the shift to mobile devices such as the iPhone and iPad is that specialist Apps designed for a single purpose will be used more than general-purpose Web browsers. A third trend from the commercial perspective is that the Internet will be dominated by a relatively small number of large players, such as Apple, who will act like the media empires of the third quarter of the twentieth century. In the article "Google: The search party is over" [80], an analysis of the differences between the stock prices and values of Google and Apple ($156 vs $236 billion, respectively) is used to support a claim that search will no longer be the most significant part of Web applications. An associated claim is that search will be supplanted by information gathering from colleagues and friends via social networking sites.

These claims are not agreed by all however. For example, in a TechCrunch article “When Wrong, Call Yourself Prescient Instead” [82], the authors cite previous predictions of the Web’s demise which proved to be false. One thing that can be assumed safely is that the debate will continue for some time. After a decade of research and as shown in the rest of this book, the Web is a global infrastructure that benefits significantly from the use of semantics. Semantics supports a broad range of tasks including data sharing and data integration at scale, knowledge management, decision making, data analysis, search, and the use and management of Web applications based on Web APIs and services, as well as a variety of vertical sectors such as government, science, business, and media. Given the success thus far, it is clear that semantic technology will also play a major role in other global network infrastructures based on, for example, mobile devices and sensor nets. Whatever form future planet-scale networks take, it has certainly been an exhilarating journey so far and we look forward to the next decade.

Acknowledgments

We thank Ian Horrocks and Michael Kifer from preventing mistakes in the sections of the chapter related to logic. We also thank Neil Benn for his help in the final formatting stages.

References

1. Adida, B., Birbeck, M. (eds.): RDFa primer: bridging the human and data webs, W3C Working Group Note (Oct 2008)
2. Antoniou, G., van Harmelen, F.: A Semantic Web Primer, 2nd edn. MIT Press, Cambridge (2008)
3. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook. Cambridge University Press, Cambridge (2003)
4. Benjamins, V.R., Plaza, E., Motta, E., Fensel, D., Studer, R., Wielinga, B., Schreiber, G., Zdrahal, Z., Decker, S.: IBROW3 an intelligent brokering service for knowledge-component reuse on the world-wide web. In: Proceedings of the 11th Banff Knowledge Acquisition for Knowledge-Based System Workshop (KAW 1998), Banff. http://ksi.cpsc.ucalgary.ca/KAW/KAW98/benjamins3/ (1998). Accessed Aug 2010
5. Berners-Lee, T.: Information management: a proposal. March 1989 and later redistributed unchanged apart from the date added in May 1990. http://www.w3.org/History/1989/proposal.html (1989). Accessed Aug 2010
6. Berners-Lee, T., Hendler, J., Lassila, O.: The semantic web. *Scientific American Magazine*, pp. 29–37 (May 2001)
7. Bizer, C., Heath, T., Berners-Lee, T.: Linked data – the story so far. Int. J. Semant. Web Inf. Syst. **5**(3), 1–22 (2009)
8. Brachman, R.J., Levesque, H.J.: Knowledge Representation and Reasoning. Morgan Kaufmann, San Francisco (2004)
9. Brickley, D., Guha, R.V. (eds.): RDF vocabulary description language 1.0: RDF schema. W3C Recommendation (Feb 2004)
10. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. Comput. Netw. ISDN Syst. **30**(1–7), 107–117 (1998)
11. Bush, V.: As we may think. The Atlantic Monthly, July 1945. http://www.theatlantic.com/past/docs/unbound/flashbks/ computer/bushf.htm (1945). Accessed Aug 2010

12. Cerf, V., Kahn, B.: "A Protocol for Packet Network Interconnection", which specified in detail the design of a Transmission Control Protocol (TCP) (1974)
13. Chen, W., Kiefer, M., David, S.W.: HiLog: a foundation for higher-order logic programming. J. Log Program **15**(3), 187–230 (1993)
14. Clocksin, W.F., Mellish, C.S.: Programming in Prolog, 5th edn. Springer, New York (2003)
15. Codd, E.F.: The Relational Model for Database Management: Version 2. Addison-Wesley Longman, New York (1990)
16. Dean, M., Schreiber, G. (eds.): OWL web ontology language reference, W3C Recommendation (Feb 2004)
17. Feigenbaum, E.A.: The art of artificial intelligence: themes and case studies of knowledge engineering. In: Proceedings of the Fifth International Joint Conference on Artificial Intelligence (IJCAI 1977), Cambridge (1977)
18. Fensel, D., Decker, S., Erdmann, M., Studer, R.: Ontobroker: the very high idea. In: Proceedings of the 11th International Florida Artificial Intelligence Research Society Conference (FLAIRS 1998), Sanibel Island, pp. 131–135 (1998)
19. Fensel, D., Angele, J., Decker, S., Erdmann, M., Schnurr, H.-P., Studer, R., Witt, A.: Lessons learned from applying AI to the web. J. Cooperat. Inf. Syst. **9**(4) (2000)
20. Fensel, D., van Harmelen, F., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F.: OIL: an ontology infrastructure for the semantic web. IEEE Intell. Syst. **16**(2), 38–45 (2001)
21. Fensel, D., Lausen, H., Polleres, A., De Bruijn, J., Stollberg, M., Roman, D., Domingue, J.: Enabling Semantic Web Services: The Web Service Modeling Ontology. Springer, New York (2007)
22. Fensel, D.: Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce. Springer, Berlin (2001), 2nd edn. Springer (2003)
23. Garcia-Molina, H., Ullman, J.D., Widom, J.: Database Systems: The Complete Book, 2nd edn. Prentice Hall, New Jersey (2009)
24. Giarratano, J.C., Riley, G.D.: Expert Systems: Principles and Programming, 4th edn. PWS, Boston (2004)
25. Gruber, T.R.: A translation approach to portable ontology specifications. Knowl. Acquis. **5**(2), 199–220 (1993)
26. Gruber, T.: Siri: a virtual personal assistant. Keynote Presentation at Semantic Technologies Conference. http://tomgruber.org/writing/semtech09.htm (2009). Accessed Aug 2010
27. Gruber, T.: Big Think Small Screen: how semantic computing in the cloud will revolutionize the consumer experience on the phone. Keynote Presentation at Web 3.0 Conference. http://tomgruber.org/writing/web30jan2010.htm (2010). Accessed Aug 2010
28. Halpin, H., Davis, I. (eds.): GRDDL primer, W3C Working Group Note (June 2007)
29. Hedman, S.: A First Course in Logic. Oxford University Press, Oxford (2004)
30. Horrocks, I.: Using an expressive description logic: FaCT or fiction? In: Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR' 1998), pp. 636–647 (1999)
31. Horrocks, I., Patel-Schneider, P.F., Boley, H., Tabet, S., Grosof, B., Dean, M.: SWRL: A semantic web rule language combining OWL and RuleML, W3C Member Submission (May 2004)
32. ter Horst, H.J.: Completeness, decidability and complexity of entailment for RDF schema and a semantic extension involving the owl vocabulary. J. Web Semant. **3**(2–3), 79–115 (2005)
33. Isaac, A., Summers, E. (eds.): SKOS simple knowledge organization system primer, W3C Working Group Note (Aug 2009)
34. Jurafsky, D., Martin, J.H.: Speech and Language Processing, 2nd edn. Prentice Hall, New Jersey (2009)
35. Kelly, J.: The Essence of Logic. Prentice Hall, New Jersey (1997)
36. Kifer, M., Boley, H. (eds.): RIF overview, W3C Working Group Note (June 2010)
37. Kifer, M., Lausen, G., Wu, J: Logic foundations of object-oriented and frame-based systems. J. ACM **42**, 741–843 (1995)
38. Lausen, H., Farrell, J. (eds.): Semantic annotations for WSDL and XML schema, W3C Recommendation (Aug 2007)
39. Lloyd, J.W.: Foundations of Logic Programming, 2nd edn. Springer, Berlin (1987)
40. Luke, S., Specto, L., Rager, D., Hendler, J.: Ontology-based Web agents. In: Proceedings of the First International Conference on Autonomous Agents (ICAA 1997), Marina del Rey, pp. 59–66 (1997)

41. Manning, C.D., Raghavan, P., Schutze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
42. Manola, F., Miller, E. (eds.): RDF primer, W3C Recommendation (Feb 2004)
43. Maslow, H.: The Psychology of Science. Harper & Row, New York (1966)
44. Mauritius National Assembly: The constitution. http://www.gov.mu/portal/AssemblySite/menuitem.ee3d58b2c32c60451251701065c521ca/. Accessed 6 Sept 2010
45. Mead, G.H.: Mind, Self, and Society. The University of Chicago Press, Chicago (1934)
46. Moens, M.-F.: Information Extraction: Algorithms and Prospects in a Retrieval Context. Springer, New York (2006)
47. Motik, B., Grau, B.C., Horrocks, I., Wu, Z., Lutz, C. (eds.): OWL 2 web ontology language profiles, W3C Recommendation (Oct 2009)
48. Nelson, T.H.: A file structure for the complex, the changing, and the indeterminate. In: Proceedings of the 20th National Conference, Association for Computing Machinery, New York (1965)
49. Patel-Schneider, P.F., Hayes, P., Horrocks, I.: OWL web ontology language semantics and abtrsct syntax. Section 5. RDF-compatible model-theoretic semantics, W3C (2004)
50. Pingdom: Internet 2009 in numbers. http://royal.pingdom.com/2010/01/22/internet-2009-in-numbers/ (2010). Accessed 6 Sept 2010
51. Prud'hommeaux, E., Seaborne, A.: SPARQL query language for RDF, W3C Recommendation (Jan 2008)
52. Reynolds, D.: OWL 2 RL in RIF, W3C Working Group Note (June 2010)
53. Robinson, A., Voronkov, A. (eds.): Handbook of Automated Reasoning. Elsevier Science, Amsterdam (2001)
54. Russell, S., Norvig, P.: Artificial Intelligence – A Modern Approach, 2nd edn. Prentice Hall, New Jersey (2003)
55. Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., Van de Velde, W., Wielinga, B.: Knowledge Engineering and Management: The Common KADS Methodology. MIT Press, Cambridge (2000)
56. Tomasello, M.: Origins of Human Communication. MIT Press, Cambridge (2008)
57. http://www.bbc.co.uk/blogs/bbcinternet/2010/07/bbc_world_cup_2010_dynamic_sem.html. Accessed 6 Sept 2010
58. http://www.bbc.co.uk/blogs/bbcinternet/2010/07/the_world_cup_and_a_call_to_ac.html. Accessed 6 Sept 2010
59. http://www.businessinsider.com/apple-buys-siri-a-mobile-assistant-app-as-war-with-google-heats-up-2010-4. Accessed 6 Sept 2010
60. http://challenge.semanticweb.org/. Accessed 6 Sept 2010
61. http://www.comscore.com/Press_Events/Press_Releases/2010/1/Global_Search_Market_Grows_46_Percent_in_2009. Accessed 6 Sept 2010
62. http://www.dagstuhl.de/en/program/calendar/semhp/?semnr = 00121. Accessed 6 Sept 2010
63. http://developers.facebook.com/docs/opengraph. Accessed 6 Sept 2010
64. http://dictionary.reference.com/browse/meaning. Accessed 6 Sept 2010
65. http://dublincore.org/. Accessed 6 Sept 2010
66. http://www.everyhit.com. Accessed 6 Sept 2010
67. http://www.fipa.org/. Accessed 6 Sept 2010
68. http://www.foaf-project.org/. Accessed 6 Sept 2010
69. http://googleblog.blogspot.com/2010/07/deeper-understanding-with-metaweb.html. Accessed 6 Sept 2010
70. http://www.ist-world.org/ProjectDetails.aspx?ProjectId=e132f5b74a41456f95611eb7ad3abfd3. Accessed 6 Sept 2010
71. http://knowledgeweb.semanticweb.org/. Accessed 6 Sept 2010
72. http://ksi.cpsc.ucalgary.ca/KAW/. Accessed 6 Sept 2010
73. http://www2.labour.org.uk/gordon-browns-speech-on-building-britains-digital-future,2010-03-26. Accessed 6 Sept 2010
74. http://www.larkc.eu/. Accessed 6 Sept 2010
75. http://microformats.org/. Accessed 6 Sept 2010
76. http://www.openlinksw.com/weblog/oerling/?id=1614. Accessed 6 Sept 2010
77. http://www.oracle.com/technetwork/database/options/semantic-tech/index.html. Accessed 6 Sept 2010
78. http://www.pandia.com/sew/383-web-size.html. Accessed 6 Sept 2010
79. http://store.levi.com/. Accessed 6 Sept 2010

80. http://tech.fortune.cnn.com/2010/07/29/google-the-search-party-is-over/. Accessed 6 Sept 2010
81. http://techcrunch.com/2010/04/21/facebook-like-button/. Accessed 6 Sept 2010
82. http://techcrunch.com/2010/08/17/when-wrong-call-yourself-prescient-instead/. Accessed 6 Sept 2010
83. http://technology.timesonline.co.uk/tol/news/tech_and_web/the_web/article7104354.ece. Accessed 6 Sept 2010
84. http://www.wired.com/magazine/2010/08/ff_webrip/all/1
85. http://WordNet.princeton.edu/. Accessed 6 Sept 2010
86. Wikipedia: Controlled vocabulary. http://en.wikipedia.org/wiki/Controlled_vocabulary (2010). Accessed 6 Sept 2010
87. Wikipedia: Energy. http://en.wikipedia.org/wiki/Energy (2010). Accessed 6 Sept 2010
88. Wikipedia: Equipment. http://en.wikipedia.org/wiki/Equipment (2010). Accessed 6 Sept 2010
89. Wikipedia: Formal_semanics. http://en.wikipedia.org/wiki/Formal_semantics (2010). Accessed 6 Sept 2010
90. Wikipedia: Idea. http://en.wikipedia.org/wiki/Idea (2010). Accessed 6 Sept 2010
91. Wikipedia:. Intention. http://en.wikipedia.org/wiki/Intention (2010). Accessed 6 Sept 2010
92. Wikipedia: Machine. http://en.wikipedia.org/wiki/Machine (2010). Accessed 6 Sept 2010
93. Wikipedia: NLS (computer system). http://en.wikipedia.org/wiki/NLS_%28computer_system%29 (2010). Accessed 6 Sept 2010
94. Wikipedia: OSI model. http://en.wikipedia.org/wiki/OSI_model (2010). Accessed 6 Sept 2010
95. Wikipedia: Purpose. http://en.wikipedia.org/wiki/Purpose (2010). Accessed 6 Sept 2010
96. Wikipedia: Second-order logic. http://en.wikipedia.org/wiki/Second-order_logic (2010). Accessed 6 Sept 2010
97. Wikipedia: Semantic HTML. http://en.wikipedia.org/wiki/Semantic_HTML (2010). Accessed 6 Sept 2010
98. Wikipedia: Semantics. http://en.wikipedia.org/wiki/Semantics (2010). Accessed 6 Sept 2010
99. Wikipedia: SLD_resolution. http://en.wikipedia.org/wiki/SLD_resolution (2010). Accessed 6 Sept 2010
100. Wikipedia:. SQL. http://en.wikipedia.org/wiki/SQL (2010). Accessed 6 Sept 2010
101. Wikipedia: Tag cloud. http://en.wikipedia.org/wiki/Tag_cloud (2010). Accessed 6 Sept 2010
102. Wikipedia: Taxonomies. http://en.wikipedia. org/wiki/Taxonomies (2010). Accessed 6 Sept 2010
103. Wikipedia: Tower of Babel. http://en.wikipedia.org/wiki/Tower_of_Babel (2010). Accessed 6 Sept 2010
104. Wiktionary: Device. http://en.wiktionary.org/wiki/device (2010). Accessed 6 Sept 2010
105. World Wide Web Consortium: OWL web ontology language reference, W3C Recommendation. http://www.w3.org/TR/owl-ref/ (Feb 2004). Accessed 6 Sept 2010
106. World Wide Web Consortium: RDB2RDF working group. http://www.w3.org/2001/sw/rdb2rdf/. Accessed 6 Sept 2010
107. World Wide Web Consortium: RDF primer, W3C Recommendation. http://www.w3.org/TR/rdf-primer/. Accessed 6 Sept 2010
108. World Wide Web Consortium: The global structure of an HTML document, W3C Recommendation. http://www.w3.org/TR/html401/struct/global.html#edef-META. Accessed 6 Sept 2010
109. World Wide Web Consortium: XML technology, W3C Standard. http://www.w3.org/standards/xml/. Accessed 6 Sept 2010
110. Yeates, G.: Earthworms. Te Ara – the encyclopedia of New Zealand (updated 1 March 2009). http://www.teara.govt.nz/en/earthworms/3/1 (2009). Accessed 6 Sept 2010