MORGAN & CLAYPOOL PUBLISHERS

The Epistemology of Intelligent Semantic Web Systems

Mathieu d'Aquin
Enrico Motta

Synthesis Lectures on The Semantic Web: Theory and Technology
Ying Ding and Paul Groth, *Series Editors*

The Epistemology of Intelligent Semantic Web Systems

Synthesis Lectures on the Semantic Web: Theory and Technology

Editor
Ying Ding, ***Indiana University***
Paul Groth, ***Elsevier Labs***

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

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

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

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

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

Semantic Mining of Social Networks
Jie Tang and Juanzi Li
2015

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

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

Semantics in Mobile Sensing
Zhixian Yan and Dipanjan Chakraborty
2014

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

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

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

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

The Epistemology of Intelligent Semantic Web Systems

Mathieu d'Aquin and Enrico Motta

www.morganclaypool.com

ISBN: 9781627051613 paperback
ISBN: 9781627050005 ebook

DOI 10.2200/S00708ED1V01Y201603WBE014

A Publication in the Morgan & Claypool Publishers series
SYNTHESIS LECTURES ON THE SEMANTIC WEB: THEORY AND TECHNOLOGY

Lecture #14
Series Editors: Ying Ding, *Indiana University*
Paul Groth, *Elsevier Labs*
Founding Editor Emeritus: James Hendler, *Rensselaer Polytechnic Institute*
Series ISSN
Print 2160-4711 Electronic 2160-472X

The Epistemology of Intelligent Semantic Web Systems

Mathieu d'Aquin and Enrico Motta
Knowledge Media Institute, The Open University

SYNTHESIS LECTURES ON THE SEMANTIC WEB: THEORY AND TECHNOLOGY #14

MORGAN & CLAYPOOL PUBLISHERS

ABSTRACT

The Semantic Web is a young discipline, even if only in comparison to other areas of computer science. Nonetheless, it already exhibits an interesting history and evolution. This book is a reflection on this evolution, aiming to take a snapshot of where we are at this specific point in time, and also showing what might be the focus of future research.

This book provides both a conceptual and practical view of this evolution, especially targeted at readers who are starting research in this area and as support material for their supervisors. From a conceptual point of view, it highlights and discusses key questions that have animated the research community: what does it mean to be a Semantic Web system and how is it different from other types of systems, such as knowledge systems or web-based information systems? From a more practical point of view, the core of the book introduces a simple conceptual framework which characterizes Intelligent Semantic Web Systems. We describe this framework, the components it includes, and give pointers to some of the approaches and technologies that might be used to implement them. We also look in detail at concrete systems falling under the category of Intelligent Semantic Web Systems, according to the proposed framework, allowing us to compare them, analyze their strengths and weaknesses, and identify the key fundamental challenges still open for researchers to tackle.

KEYWORDS

semantic web, linked data, intelligent systems, knowledge engineering, knowledge-based systems, ontologies

Contents

Preface

In this book, we take both a conceptual and a practical view of the Semantic Web. The Semantic Web is a young discipline, even if compared only to other areas of computer science. Nonetheless, it already exhibits an interesting history and evolution. From a very general vision which, as presented in the foundational article by Berners Lee et al. (2001), integrated different techniques and approaches (agents, databases, knowledge representation, web technologies, etc.), the focus quickly moved to establishing and standardizing a core set of technologies for the representation, distribution, and access to knowledge and data on the Web—e.g., RDF, OWL, SPARQL, etc. Once these core technologies were established, the research community was then able to focus on developing applications and building a large-scale web of data, in accordance with the Linked Data principles.

Having gone in a rather accelerated manner through this cycle, from vision to impact, this is the time to see where this is all going. This is the conceptual aspect of this book: The *Epistemology* of Intelligent Semantic Web Systems. The debate that started as soon as the idea of the Semantic Web was put forward by Tim Berners Lee and colleagues, mostly from the Artificial Intelligence area, has indeed never stopped: What does it mean to be a Semantic Web system? How is it different from other types of systems, such as knowledge systems or web-based information systems? Our goal is not necessarily to provide definitive answers to these questions, but to highlight and discuss in a concise manner the key elements that need to be considered and understood when engaging with this new class of systems.

First, we look at the Semantic Web as part of the Web, and characterize it as a conceptual construct, part of a stack of structures and networks, rather than a stack of technologies (Chapter 1). The idea here is that, by understanding what makes the Semantic Web something of a higher level of abstraction than the Web itself, we have a better view of the implications in terms of the challenges to be addressed, and of the opportunities for new applications and systems to be developed. This view of the Semantic Web naturally leads to thinking of it as a distributed, networked system where knowledge is shared and managed globally. It also leads us to think of it as a type of knowledge system with specific characteristics to do with being distributed, networked, and thus open and without centralized control.

Second, we consider conceptual models established in the field of Knowledge-based Systems and explore how they can be adapted to the open, decentralized nature of the Semantic Web (Chapter 2). In doing so, we establish a simple conceptual framework to characterize Intelligent Semantic Web Systems in terms of the functions and components they include. The goal here is not only to better understand what makes an Intelligent Semantic Web System, but also to establish a reference model useful both to recognize and compare Intelligent Semantic Web

Systems, and also to guide their design. This framework can be seen as defining the core of the book, joining up the conceptual discussion on the nature of the Semantic Web with the more practical view on what it means to develop an Intelligent Semantic Web System. In contrast with other characterizations, here we do not focus on actual technologies but on the various conceptual elements that need to be present when designing Intelligent Semantic Web Systems.

The practical aspect of this book is therefore in this focus on Intelligent Semantic Web *Systems*, and in the deliberately rapid jump from debating the nature of the Semantic Web, to looking at what it means concretely. Hence, once we have established the general, conceptual framework, we look in detail at concrete systems that, according to this framework, fall into the category of Intelligent Semantic Web Systems (Chapter 3). The goal here is that, by looking at concrete examples, we can see how the general model captured by the Intelligent Semantic Web System framework is instantiated in practice. In addition, the model also allows us to compare different systems, thus analyzing their relative strengths and weaknesses and their conformance to our framework.

Having clarified the nature of Intelligent Semantic Web Systems, we are then left with the question: "What next?" Hence, in the final chapter, we look at the key open research challenges that come out of our analysis. To some extent, here we come to two seemingly contradictory conclusions: On the one hand, the area of Intelligent Semantic Web Systems has evolved into a mature field with research directly leading to major opportunities and impacts already clearly visible in many academic and commercial systems. On the other hand, the area is still in its infancy with many fundamental challenges still open for researchers to tackle.

This contradiction is one of the reasons why this book is especially suitable for those (e.g., students) who are starting research in this area and as support material for their supervisors. Much of the material in this book comes from our own experience of more than a decade working as researchers and application developers in the Semantic Web area and from the many discussions on these issues we have had with other members of the research community (see for example d'Aquin et al., 2008a). Much of the thinking here also comes from the experience of establishing in 2003 the first international summer school dedicated to Semantic Web Research (SSSW). SSSW is still today the key educational event in the Semantic Web area and has evolved dramatically since its first edition, reflecting the evolution of the area but also, to some extent, contributing to shaping it. This book is a reflection on this evolution, with the aim to take a snapshot of where we are at this specific point in time, and also to show what the future will be like, or at least should be like!

Mathieu d'Aquin and Enrico Motta
April 2016

Acknowledgments

The content of this book is based on research started by our group more than a decade ago. It derives, directly or indirectly, from the work of the past and present members of this group, namely: Alessandro Adamou, Sofia Angeletou, Elizabeth Cano Basave, Carlo Allocca, Claudio Baldassarre, Emanuele Bastianelli, Enrico Daga, Maurizio di Matteo, Martin Dzbor, Salman Elahi, Miriam Fernandez, Jorge Gracia, Laurian Gridinoc, Davide Guidi, Tom Heath, Yuangui Lei, Ning Li, Shuangyan Liu, Vanessa Lopez, Andriy Nikolov, Francesco Osborne, Michele Pasin, Silvio Peroni, Dnyanesh Rajpathak, Marta Sabou, Angelo Antonio Salatino, Lucia Specia, Keerthi Thomas, Ilaria Tiddi, Victoria Uren, Maria Vargas-Vera, Paul Warren, Fouad Zablith and many others who have been, in one way or another, associated with us. We are immensely grateful to have had the opportunity to work with such brilliant researchers and to debate with them the state, direction, and impact of the semantic web, linked data, and Intelligent Semantic Web Systems. We are also grateful to all our colleagues, who came for a visit or we met at conferences, seminars, and other events, and with whom we discussed and wrote about the challenges and opportunities of the semantic web. Finally, we thank our families and friends for being there and mostly bearing with us.

Mathieu d'Aquin and Enrico Motta
April 2016

CHAPTER 1

Characterizing the Semantic Web

The Semantic Web, and by extension semantic web technologies, is very young in comparison to other computing disciplines, such as databases and artificial intelligence—and indeed even the Web is very young in comparison with these disciplines.[1] As a result, as is usually the case with new phenomena, it will probably take time to develop a comprehensive understanding of the Semantic Web and distinguish its fundamental aspects from the purely coincidental ones. Hence, it is no surprise that defining the Semantic Web has been, in the past dozen years of active research in the area, both a difficult thing to do, and an evolving exercise. For example, many initial solutions in this area put a lot of emphasis on the "semantics" aspect, giving an interpretation to the word that relates it to the long standing area of artificial intelligence and (logic-based) knowledge representations [Genesereth and Nilsson, 1987]. Taking this view, the Semantic Web has been characterized as a web in which information is interpreted and reasoned upon by software agents acting on our behalf [Berners-Lee et al., 2001]. The key here is the association of web documents with formal semantics, expressed by means of logical models encoded in web *ontologies* [Staab and Studer, 2009]. The ability to achieve logical inferences on such models was seen as the true hallmark of a *Semantic* web, with the *web* aspect of considering the distribution of knowledge and information in a global, collaborative network being left in the background.

This logicist stance has of course led to very valuable work, producing standards for expressing formal semantics in ontologies [Horrocks et al., 2003], and also inference systems able to scale to vast amounts of semantically characterized data (e.g., see Sirin et al., 2007). As a result of adopting this perspective, many of the early Semantic Web applications were actually rather similar to earlier knowledge-based systems, with an additional emphasis on web interfaces and knowledge sharing and reuse [d'Aquin et al., 2008a, van Harmelen et al., 2009]. Specifically, this similarity was primarily a result of focusing more on the "semantic" rather than the "web" aspect, and emphasizing therefore the ability to reason with formal representations of knowledge, rather than the "web-like" ability of operating in an open and distributed world [d'Aquin et al., 2008a].

Other characterizations of the Semantic Web have instead focused on the technological elements. When considering the Semantic Web as a platform for the global exchange of machine

[1] The origin of the semantic web is generally associated with the article Berners-Lee et al. [2001], and the start of the Web can also be linked to an article (a proposal) written by Tim Berners-Lee in 1989 (see `http://www.w3.org/History/1989/proposal.html`), while database systems were already an established field of study in the 1960s and the first artificial intelligence conference took place in 1956 (see Russell and Norvig, 2003).

readable information, and given that the Web has only been made possible through the adherence to standard technologies, it is natural to characterize the semantic web also through a stack of standard technologies [Horrocks et al., 2005]. In such a view, a semantic web system is one which uses RDF, URIs, and HTTP to model and share information, and which reuses standard vocabularies (i.e., ontologies) to structure data in such a way that they can be reused by applications, using web technologies as a medium for their distribution. By and large, this is the view at the core of the linked data movement [Bizer et al., 2009], which emphasizes the open publication of data using standard web technologies and the use of the architecture of the Web (URIs and links, see below) for the representation of data.

Our view is that both these perspectives are unsatisfactory (for rather different reasons) and we will therefore try to elaborate what we believe is the essence of this new technology that is the semantic web: how "being a web" makes it fundamentally different from traditional knowledge-based systems and how the technologies employed are merely a reflection, or an instantiation, of more fundamental properties. Of course, this is not purely done as an academic exercise (even if it is an interesting one). By doing so, we hope to clarify what are the fundamental characteristics of intelligent semantic web systems, regardless of the languages and technologies they use, and how they can be understood as part of a new discipline at the boundaries of different fields, including software engineering, artificial intelligence, web development, data processing, human-computer interaction, and others.

1.1 IT IS A STACK AFTER ALL!

To achieve a more fundamental understanding of the semantic web, we need to consider first what is the Web. Naturally, it is tempting to describe the Web by highlighting the underlying technology as the core defining element of the concept: "the Web is whatever uses HTTP over a network." Here however, the more conceptual view might be as easily expressible: "The Web is a graph of documents connected by hyperlinks."

This might appear too simple to do justice to the importance and impact of the Web, but what is really interesting is the implications of such a basic conceptual definition. Documents on the Web have web addresses (URIs) and connecting them is made by simple reference to these web addresses. This means that the network formed by the Web is a purely conceptual network that transcends the physical network and the software infrastructure on top of which it is sitting. To say it plainly, the power of the Web is that it does not matter where, by whom, and using which tool a document was created and published on the Web, for it to be part of the network and connected to the rest of the global space of web documents. This is achieved because the Web abstracts from these considerations and assumes they are dealt with appropriately by means of lower level technologies.

Considering this, it is quite natural to envision the semantic web similarly as a stack, where higher level concepts abstract from the lower level, more concrete realizations. This has been considered in particular through the well known "semantic web technology stack" (see Figure 1.1)

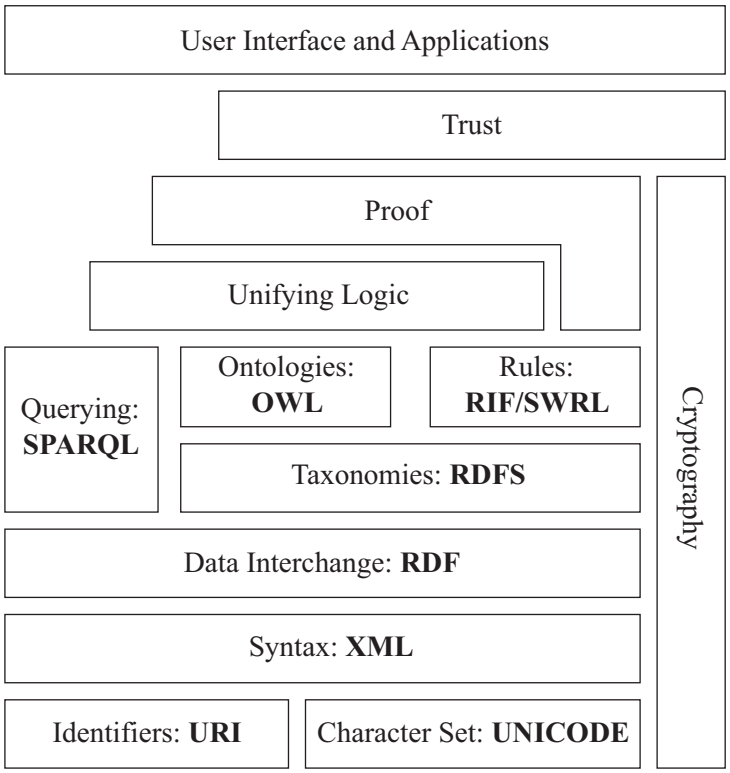

Figure 1.1: One of the instantiations of the Semantic Web Technology Stack (from http://en.wikipedia.org/wiki/Semantic_Web_Stack).

first mentioned in Berners-Lee [2003] and further refined by numerous authors (e.g., Horrocks et al., 2005). There is however something fundamentally unsatisfactory about using this stack to explain and define what is the semantic web, and in particular trying to characterize what kind of intelligent systems can be developed as semantic web applications. The reason is that this is simply a stack of "accidental" technologies, describing how existing technologies (URIs, RDF, etc.) rely on each other, and how future standards might eventually rely on these elementary technologies. The stack does not explain the concepts that define the essence of the semantic web, but (possible) relationships between components, which might be used to implement it. For example, the mention of URIs in this stack is a reflection of the need for the Semantic Web to include a mechanism for expressing global, universal identifiers for data objects and abstract concepts. URIs might be the best (and possibly only) candidate for this, but it is only one instantiation of this concept. Similarly, RDF in this stack is an instantiation of the notion of a distributed, graph-based data model, while RDFS is simply the schema model for such a data model.

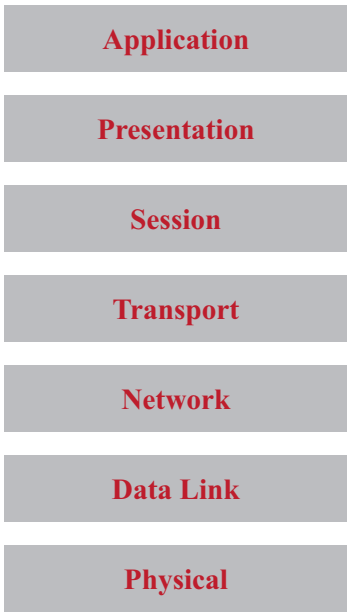

Figure 1.2: The OSI model—networking layers.

Nonetheless the idea of characterizing the Semantic Web as a stack of ever more abstract notions is appealing. Indeed, we can take inspiration from the Open Systems Interconnection (OSI) stack used to categorize networking technologies (Zimmermann [1980]; see Figure 1.2). Here, the idea is that each layer manipulates objects and notions at a higher level of abstraction than the one below and simply assumes that the layer underneath is able to fulfill its role in the process of transferring information. The physical layer is about cables, signals, and hardware, the data link layer about reliably transferring data from one machine to another through a direct connection, the network layer about transferring data from one point of the network to another, the transport layer about delivering packets of information between nodes of a network, the session layer about maintaining connections, the presentation layer (mostly) about the representation of information in a machine-independent way and the application layer about whatever might be done with the ability to transfer information from one (virtual) place to another. Each of these features might be implemented through a number of different technologies, achieving the same purpose in different ways. For example, the physical layer might be achieved through Wifi or Ethernet or, at transport level, one might use TCP or UDP. Conversely, each layer also encapsulates elements of the higher level layers, without needing to process or understand them. Indeed, the physical layer for example only assumes bits of data to be transferred, and transforms these bits into signals. Similarly, the transport layer (e.g., TCP) does not need to interpret the content

of the information to be transferred from the layers above (e.g., emails, web pages, etc.), and only needs to care about implementing a protocol to get any arbitrary packet of information from one node to the other. In the OSI stack, the Web belongs to the top layer, called (somewhat misleadingly in this case) the application layer, which means that the Web abstracts from the physical and implementation details that require information to be located and transferred between different places, machines, and systems.

The Semantic Web has been described as "an extension to the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [Berners-Lee et al., 2001]. Similarly, linked data can be seen both as relying on the basic infrastructure of the Web for the purpose of data sharing, and also as a pragmatic reduction of the original Semantic Web vision. While the vision of the Semantic Web was essentially about giving meaning to information on the Web, linked data is essentially about a pragmatic way to represent and deliver such information as a true part of the Web (using URIs and web links as the basis for a global graph data model, see Bizer et al., 2009). Based on this simple line of reasoning (i.e., the Semantic Web as a layer defined over the Web), we can then consider the Semantic Web as part of a stack, which defines a set of ever more abstract notions. Specifically, we start with the basic mechanisms for the encapsulation and transmission of information in documents (the Web), we move then to the materialization of this information as processable and interconnectable data entities (linked data), then to the interpretation of this information through the application of semantics (the semantic web)—see Figure 1.3.

This characterization can be directly related to the one separating the "symbol level" from the "knowledge level" in knowledge representation and knowledge-based systems [Newell, 1982] and indeed it can be seen as the adaptation of the classic analysis from Newell to the new scenario emerging with the semantic web. The symbol level (here web technologies and linked data) is system oriented and mostly concerned with the mechanisms used to represent and manipulate information. The knowledge level (here the semantic web) is concerned with the more abstract understanding of this information, especially the conclusion that an agent can draw from it, independently of the way it is represented. The lower layers (the symbol level) of this new semantic web stack are quite clearly defined and have been analysed already in detail in the literature. Hence, in this book, we essentially focus on the higher levels of the stack. Specifically, we are interested in analyzing the class of intelligent systems that can be designed by taking advantage of the distributed knowledge available through the semantic web, which in turn relies on the distributed, structured information layer provided by linked data.

In the next sections, we analyze these different layers in some detail. However, in contrast with earlier analyses (e.g., Antoniou and van Harmelen, 2008, Heath and Bizer, 2011, Hitzler et al., 2011) that focused on specific technical aspects of the Semantic Web (such as the common information and knowledge representation formats/standards/practices), we strive to keep the discussion independent from a purely technological perspective (even though we will refer to

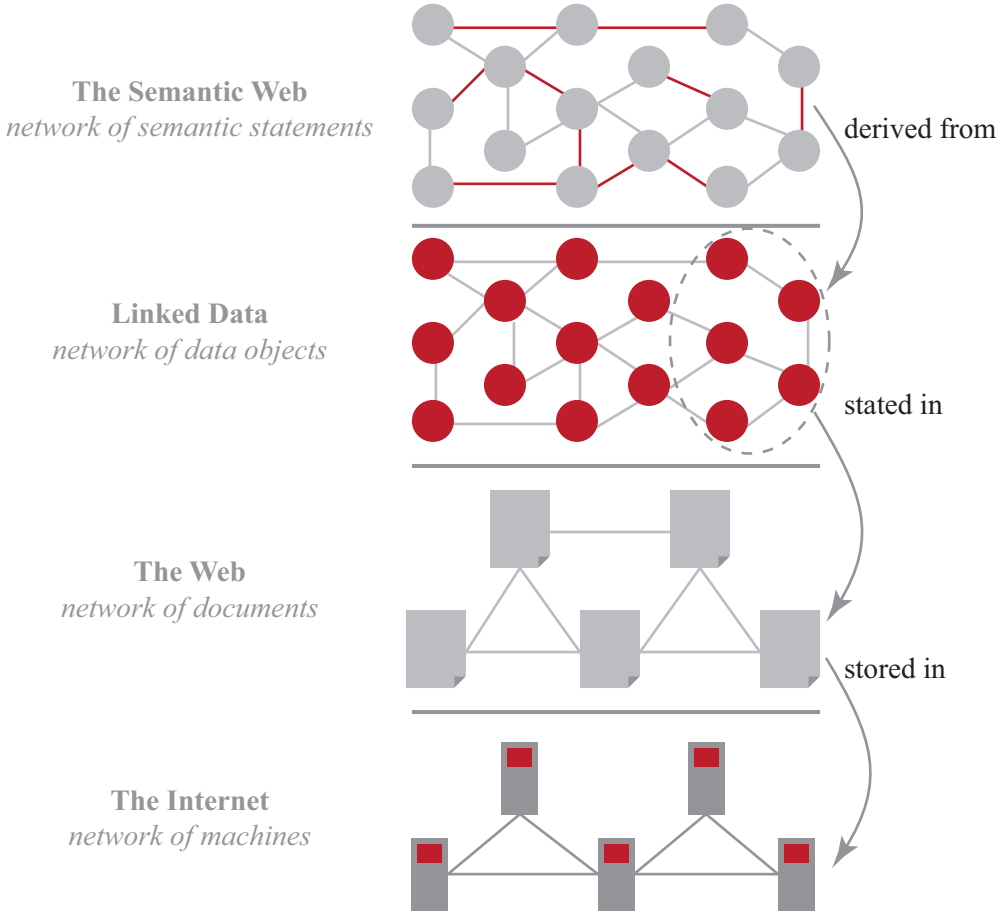

Figure 1.3: Layers of the Semantic Web.

current implementations) and focus instead on the abstract functionalities that the different layers are meant to realize.

1.2 THE WEB AS A LAYER ON TOP OF THE INTERNET

As discussed above, there may be a lot of different aspects that can be argued as being central to the definition of the semantic web. This "faceted" view on a technological platform actually already starts with the Web. Indeed, one might for example consider the Web from the point of view of networking technologies. It has already been mentioned that web technologies appear in the upper, application part of the OSI network layer stack. What this means is that *whatever the Web does* is encapsulated down into actual networking technologies for data transfer, machine-to-machine

communication, etc. The layer just below the Web in this case is the internet (materialized through the IP protocol) which is itself an abstraction from the more concrete and technologically constrained lower layers (the internet literally being a network of networks, which might be very heterogeneous from the point of view of the bottom layers).

From another point of view, the Web can be described as a hypertext system [Conklin, 1987], i.e., a system made of documents that link into each other through hyperlinks. The Web, from this point of view, can be seen as a particular instantiation of a hypertext system, having specific properties (e.g., no centralized control and consistency requirements) and relying on specific technologies and protocols, most notably HTTP [Fielding et al., 1999].

What is interesting however is that the Web can be seen as both things at the same time: the realization of a hypertext system that sits on top of global networking technologies. Hence, while following a link on the Web is concretely translated into (or encapsulated into) sending a request to a specific server, in a specific network on the internet, this mechanism is abstracted into the notion of accessing a document that is conceptually connected to another document in a way that is not affected by the technological realization of this conceptual link. This creates a global network of documents which connect across networks, machine, systems, organizations, and locations.

1.3 LINKED DATA AS A LAYER ON TOP OF THE WEB

The question of the relationship between linked data and the semantic web is a difficult one to answer, as the area has clearly spun off from the semantic web research community and many researchers may consider linked data as the concrete realization of a semantic web [Bizer et al., 2009]. Here we propose however a novel characterization of linked data, as a lower-level layer of the semantic web, which provides the basis for knowledge to be distributed, networked, and shared based on the principles that the linked data community has put forward. We first however discuss what makes linked data principles so interesting and successful, and we characterize them as "using the architecture of the Web for data linking."

Indeed, besides the technological considerations of representation languages, query mechanisms, and storage facilities, the basic idea of linked data is to rely on the same principles that have allowed the web of documents to become such a globally significant system, for the purpose of enabling the sharing of information entities rather than documents. Analogously to the Web, the fundamental element here is the identification of these information objects using web addresses (URIs) and their connection through hyperlinks. Taking a simple example (see Figure 1.4), such a linked data entity can be used to represent a person (Mathieu), which would then be assigned a URI to identify him (e.g., `http://data.open.ac.uk/person/0e5d4257051894026ea74b7ed55557e7`). Another information entity can be a particular publication, such as d'Aquin et al. [2005], which is also associated with a URI (`http://data.open.ac.uk/oro/43798`), or the higher-education institution where Mathieu is working, The Open University (`http://education.data.gov.uk/id/school/133849`). Crucially, as in the case of the Web, these

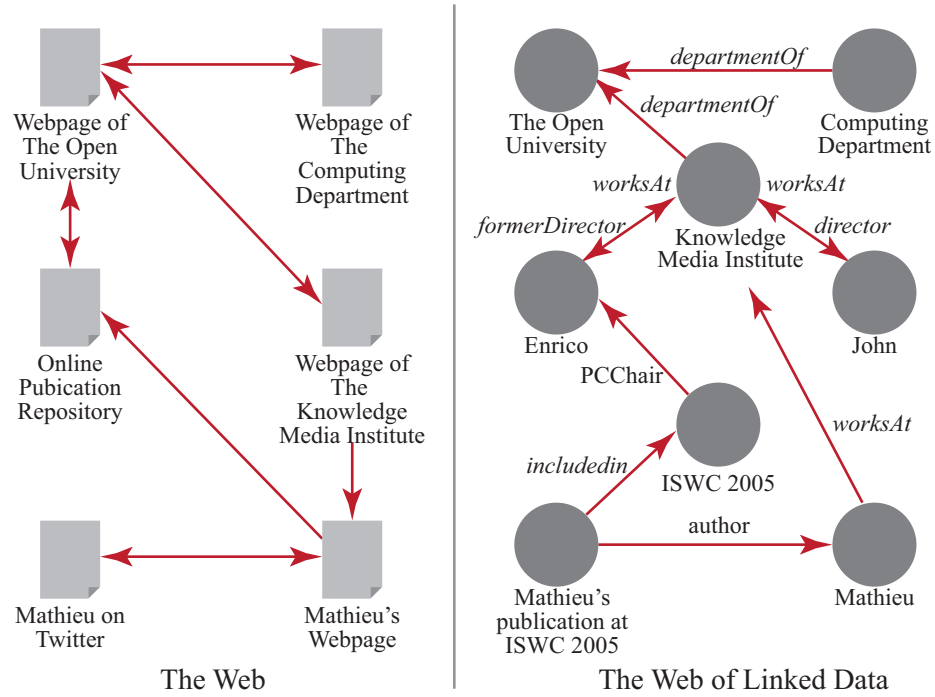

Figure 1.4: Linked data as the application of web principles to information objects.

different identifiers are not produced and maintained centrally but in a distributed fashion, by different data owners. Links can then be built between such information objects, the linked data entities, in ways similar to the hyperlinks of the web of documents. However, a crucial difference here is that such data entities are also labeled by the type of relationship that relates them. Thus, one can declare that Mathieu is one of the *co-authors* of the publication and an *employee* of the Open University. In other words, the basic mechanisms of the Web—universal identifiers (URIs), a simple protocol for obtaining resource representations from these identifiers (HTTP) and representation formats for resources that can embed links to other identifiers—are used both to publish and to model the data in a globally accessible graph.

These basic principles might seem simple, and naturally much debate has taken place with respect to their technological realization. They are however truly the essence of what makes linked data such a powerful and successful paradigm: the essential property of being independent and abstracting from the artificial boundaries and accidental connotations created by technology. In the case of the Web, one does not need any special access to declare a web link between her web documents and those of others. Here, information about Mathieu and his publications sits in a different information system (`data.open.ac.uk`) from what concerns The Open University as

an organization (data.gov.uk). Still, all these information objects exist and can be used within a single common, collective data graph where these differences are abstracted and removed [Heath and Bizer, 2011].

This is the reason why we consider linked data as a layer on top of the Web, in the same way as the Web is a layer on top of the internet, and internet protocols form a layer on top of lower level network protocols. Data modeling and sharing is encapsulated within the basic web mechanisms designed to publish and connect documents (i.e., information containers). Information about different linked entities are "put online" as web documents and through mechanisms that concern the functioning of web servers. However, at the conceptual level of linked data, these considerations become irrelevant: these are just the lower level mechanisms that make it possible to create such a network of interconnected information objects.

Much effort has been spent on extending and adapting the basic mechanisms to access documents on the Web, so that they work consistently with data on the Web of Data. RDF [Lassila and Swick, 1999] is an obvious example, building on web standards such as XML and using URIs and links as the basis for a graph-based data modeling and representation language. An even better illustration of the need for new ways to access this higher level dimension of the Web is provided by the intensive discussions around the mechanisms through which web servers should deliver linked data, how software agents should request such linked data, and how they should interpret the response [Ayers and Völkel, 2008, Heath and Bizer, 2011].

Another significant challenge in the development of linked data, also related to the need for new technological developments, is about supporting access to the data by human users, in particular with respect to providing ways to browse and present the information content. However, many "Linked Data Browsers" suffer from relying on a rather naive approach to reproduce (rather than to abstract from) the mechanisms of the Web [Karger and Schraefel, 2006]. Hence, we believe that there is still a need to better understand how to interact with information embedded in such a large, heterogeneous, and open data graph, beyond the metaphors used for somehow simpler interactions with documents.

We will come back to these issues in later chapters, through providing a more conceptual, "knowledge level" view of the way intelligent semantic web systems interact with information on the semantic web.

1.4 THE SEMANTIC WEB AS A LAYER ON TOP OF LINKED DATA

The stack model we are considering here is a way to answer the question of the relationship between linked data and the semantic web, i.e., linked data provide an intermediary step, or layer, between the web and the semantic web. As discussed earlier, the linked data layer considers elements related to the distribution of information, the integration of information, and its publication for sharing online, as encapsulated within the architecture of the Web. The semantic

web layer instead considers notions of a higher degree: how such information is interpreted and processed within a network of knowledge entities, rather than as raw data.

Such notions and considerations have been the focus of the research community on the Semantic Web since the early days and quickly materialised into specific technologies. In particular, ontologies [Staab and Studer, 2009] were chosen as the main paradigm for the representation of knowledge on the semantic web, with languages such as OWL [McGuinness and van Harmelen, 2004] devised to enable the use of formal knowledge representation formalisms (namely description logics; Baader, 2003) to encode distributed, sharable knowledge on the Web. While the aspects of data representation were not a strong focus in this view of the semantic web, the idea that we are promoting here, that the Semantic Web is a layer above linked data, is still clearly illustrated by the way such technologies are being used nowadays: the ontologies are used as a semantic structure above the data, relating them to formal notions which make it possible, to a certain extent, to manipulate them at a more abstract level of meaning than the one defined by the raw linked data entities. Following up from our example in Figure 1.4, each data object in the linked data graph might be associated (typed, labeled) with abstract concepts from web ontologies (e.g., Mathieu is a Person, The Open University is an Organization/Educational Institution/University). Similarly, the relationships used to connect these data objects can be defined as part of ontologies, themselves shared and published using the same web mechanisms as the more concrete linked data (e.g., the relation "worksAt" is defined, at a certain URI, as a relationship between a person and an organization). The role of ontologies here is therefore seen as both providing a structure for the data (for agents to know what to expect from these data) and a shareable, logical expression of the intended meaning of the concepts which data objects instantiate.

Generalizing from this, we can see that the relationship between the semantic web layer and the linked data layer provides a much steeper abstraction step than the one between linked data and the web. Indeed, it is the same abstraction step which has classically been characterized as moving from information to knowledge, where the latter is viewed as information interpreted and integrated in such a way that it may lead to the production of more knowledge, through inference [Aamodt and Nygard, 1995]. This notion of inference is actually central to the notion of semantics as understood, originally, in knowledge representation and later in the initial views on the Semantic Web that led to the focus on ontologies. In some ways, it is what makes the Semantic Web a layer above linked data, as the central idea of the Semantic Web is to make explicit, through more or less complex inferences, the knowledge which is encapsulated within the data sources integrated and made available through linked data.

Now, even if ontologies (and therefore description logics; Baader, 2003) have been a clear focus of semantic web research in the last decade, this idea that the Semantic Web represents the knowledge level (in the sense of Newell, 1982) of the (symbol-level) linked data is not only materialised through ontological and logical inferences. More generally, we take the view consistent with the motto "A little semantics goes a long way" [Hendler, 2007], that the Semantic Web is as

much materialised by simple mechanisms such as interpretable links between linked data entities, basic taxonomies, lightweight but shared schemas or analytical processes, as it is from complex, formal and logical mechanisms, such as the ones found in description logics. In other words, in the semantic web environment, simple inferences at a very large scale can be more valuable than complex inferences in a closed system [d'Aquin et al., 2008a].

Having established this principle does not however help us to solve our main challenge: how do we build intelligent systems that benefit from the semantic web? Being a level of abstraction above linked data means that the issues regarding the way in which human users and software agents interact with the Semantic Web become even more prominent here. The web paradigm of interacting with information through browsing documents becomes irrelevant in an environment made of a large, global network of inferable information. In the next chapter, we turn to the large body of work done in the area of knowledge-based systems, and especially to the ways in which knowledge-based systems have been abstracted from the base technologies used to implement them, in order to characterize them through more abstract notions, such as knowledge and symbol levels, as well as the TELL and ASK protocol [Lakemeyer and Levesque, 2000]. The straightforward approach here would be to think of the development of intelligent systems on the semantic web as creating knowledge-based systems where the Semantic Web is the knowledge base. As we will see, this view is actually an interesting starting point, as long as we take into account the challenges arising from the open nature of the Semantic Web, in contrast with the relatively closed environment of knowledge-based systems.

CHAPTER 2

Anatomy of an Intelligent Semantic Web System

The semantic web, which we have characterized in the previous chapter as a conceptual network two levels of abstractions above the web, requires new approaches and tools to enable users and applications to interact with and exploit something that is akin to a knowledge network, rather than being simply a network of documents. Knowledge-based systems have been a primary topic in artificial intelligence research for several decades.[1] Their main focus has been the encoding and use of formalized knowledge in systems that can support human users in their tasks. They have seen a wide range of applications including expert systems and decision support systems in areas such as medicine, the law, and business. Therefore, in order to better understand how to address the problem of characterizing the way automatic processes and human users can work with encoded knowledge from the semantic web, we first discuss how the question of interacting with knowledge has been addressed in knowledge-based systems and we then attempt to adapt these notions to the new wider context of the semantic web, where such knowledge is distributed over the web.

2.1 KNOWLEDGE-BASED SYSTEMS

Generally, speaking, a knowledge-based system can be described as a computing system that relies mostly on three components:

1. A *knowledge base*, encoding a representation of the knowledge and information necessary to the realization of the intended tasks,

2. an *inference engine*, which processes and queries the knowledge base to reach the necessary conclusions, and

3. an *interface*, which enables the user to interact with the system in order to query it, make use of its functionalities, and integrate it with other components [Shadbolt et al., 1993].

There is a long history of knowledge-based systems, including experts systems in many different domains, making use of a wide variety of knowledge representation and reasoning mech-

[1]We can see for example that formalizing knowledge is a central aspect of McCarthy and Hayes [1969], and remains central in many of the more recent references on Artificial Intelligence, such as Russell and Norvig [2003].

anisms. However, what fundamentally distinguishes knowledge-based systems from other systems in software engineering is summarized in the first sentence of Sowa [1983]: "Knowledge-based systems emphasize meaning." More pragmatically, as extensively discussed in that book, knowledge-based systems are an approach toward the building of intelligent computer systems that rely on the declarative expression of the knowledge required to achieve a task and of the rules that govern inferences upon such knowledge, rather than on the procedural aspects of the way the task can be achieved.

There is an interesting parallel between this notion of knowledge-based systems and the discussion related to the relationship between the Semantic Web and the "standard" web of the previous chapter. Indeed, in both cases, they represent a paradigm of building information systems that abstract from representing the "how" (how to achieve a task, how to present information in documents) into representing the "what" (knowledge and inference rules underlying the tasks, integrated data, and ontologies). In both cases, they represent higher-level extensions/generalizations of their counterparts (software systems and the web); in the case of knowledge-based systems, focusing on dedicated software applications, and in the case of the semantic web, on the networked aspects of information.

Another interesting parallel to draw between knowledge-based systems and intelligent semantic web systems is how, while they are defined at a highly conceptual level, much research in both cases has focused on the specific technologies required to implement them. That is, if we consider Newell's distinction between the *knowledge level* (what a system "knows") and the *symbol level* (how this knowledge is represented) of a knowledge-based system [Newell, 1982], we can say that much work related to the Semantic Web and intelligent semantic web systems have focused on the *symbol level*: the way information and knowledge is encoded, represented, integrated, and processed, with relatively little work so far on understanding the conceptual, formal, and abstract properties of intelligent semantic web systems [d'Aquin et al., 2008a, van Harmelen et al., 2009].

2.2 INTERACTING WITH KNOWLEDGE IN KNOWLEDGE-BASED SYSTEMS

The book *Logic of Knowledge Bases* [Lakemeyer and Levesque, 2000] addresses the question of the manipulation of knowledge in knowledge-based systems in a general sense, i.e., at the knowledge level, abstracting from specific technologies and approaches. The authors define the elementary operations which form part of the interaction language of the knowledge base, which is used to communicate knowledge to other system components (see Figure 2.1). These operations communicate through statements in the considered knowledge representation language, posing queries and making assertions, thus affecting what is referred to as the "epistemic state" of the system. This epistemic state has a very important role both in the formal definition, as an abstract object, and concretely, as it encapsulates the current state of knowledge in the system. It defines what is currently being "represented" by the knowledge base, i.e., a formal, abstract object which refers to

the knowledge of the system, on the basis of the underlying representation formalism and inference rules. To simplify, we will consider here the epistemic state and its concrete representation in a knowledge base as equivalent.

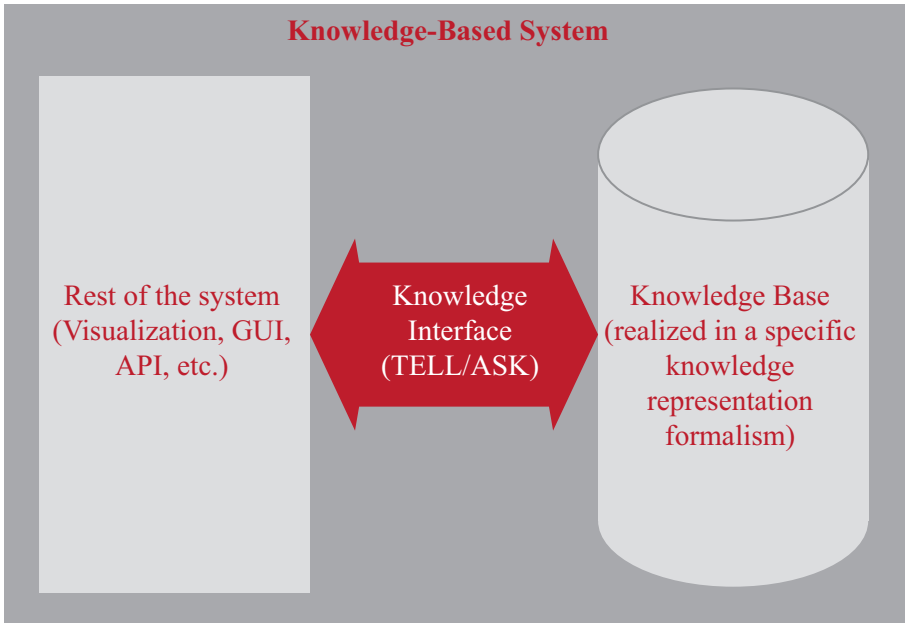

Figure 2.1: Crude overview of interacting with knowledge in a knowledge-based system.

Crucially, this view of knowledge-based systems emphasizes the role of knowledge in the system as a separate, bounded entity that the rest of the system should interact with, following a given interaction protocol. The definition of such an abstract protocol is an interesting exercise, since, as extensively discussed in Lakemeyer and Levesque [2000], it can be based on a small set of generic operations: The ones dedicated to changing the epistemic state of the system (the TELL operation, updating the knowledge base with new knowledge) and the ones dedicated to querying the epistemic state of the system (the ASK operation, to retrieve information from the knowledge base, via the inference system).

Based on Lakemeyer and Levesque [2000], we refer to these operations as the TELL/ASK protocol, as illustrated in Figure 2.1. It is worth mentioning that this abstract protocol has provided the basis of several initiatives standardizing interfaces to knowledge-management systems, including KQML (Knowledge Query and Manipulation Language—Finin et al., 1994) and DIG (Description Logic Implementation group—Bechhofer et al., 2003). In the context of our analysis, the interesting question is whether this approach is applicable to the semantic web. In other

words: Can we see semantic web applications as knowledge-based systems, where the Semantic Web is characterized as a global (although distributed) knowledge base?

2.3 INTERACTING WITH THE SEMANTIC WEB AS A KNOWLEDGE BASE

One major difference between knowledge-based systems and semantic web applications can be expressed as the major difference between a knowledge base and the semantic web: the assumption in knowledge-based systems is that knowledge is completely and finitely accessible within a closed and bounded artefact: the knowledge base. In other words, a knowledge-based system is actually a system, with clearly defined boundaries, which can be associated an epistemic state (the epistemic state of the system, as encoded by its knowledge base). However, this model cannot be applied to the Semantic Web without adaptation. As explained in Chapter 1, the Semantic Web is a knowledge network. While finite and entirely addressable in theory, the reality is that the Semantic Web cannot be manipulated as a complete and bounded entity. One notion that makes this difference evident is the one of consistency of a knowledge base, i.e., the assumption that formal knowledge within the system does not contradict itself. While knowledge-based systems do not necessarily assume the knowledge base to be consistent, they rely extensively on the fact that there is such a characteristic as the "consistency of the knowledge base." If we were to apply the same notion to the semantic web, it would become quickly evident that it is simply inconsistent, that there is nothing we can do about it, and that the extent and scope of inconsistencies cannot even be assessed. Of course, the fact that this statement sounds very unhelpful is symptomatic of the notion that it simply does not make sense: The Semantic Web cannot be seen as a bounded artefact upon which characteristics such as consistency universally apply. There is no such thing as the epistemic state of the semantic web.

There are different ways in which we can try to address this issue. One of these approaches is to think of the Semantic Web as a network of knowledge bases, where the basic characteristics of knowledge bases (including consistency) are applicable locally, but not globally. This approach is materialised for example in Suárez-Figueroa et al. [2012] through the notion of networked ontologies. In this case, each ontology in the network might represent a specific context or situation and be valid within this context or situation, and it might link to other ontologies in such a way that what is valid in the local context from the external ontologies can be integrated. Several formalisms have been proposed to support such a notion of networked ontologies, including Distributed Description Logics [Borgida and Serafini, 2003], C-OWL [Bouquet et al., 2003], or E-Connections [Kutz et al., 2004]. While these formalisms implement very sophisticated constructs to create, manipulate, and reason upon ontology networks, this perspective remains very simple: Since we cannot see the whole Semantic Web as a knowledge base, we consider it as a collection of (linked but independent) knowledge bases. This however implies that the epistemic state of a system that exploits a networked ontology is necessarily restricted to the initial arbitrary boundaries of the containers of these semantic web knowledge bases. Aside from the

problem that these boundaries are difficult to define (i.e., deciding where a connected ontology or a linked dataset starts and where it stops is not always straightforward), this contradicts the view expressed in Chapter 1 that the Semantic Web should actually be defined as an abstraction from the "symbol-level" representation of knowledge, including the ontology designer's decision to encapsulate pieces of knowledge together within the "container" of an ontology. Another issue is that it remains unclear, despite the work in Suárez-Figueroa et al. [2012], what are the role, form, and effect of the connections between such knowledge bases. In sum, it appears that thinking of the Semantic Web as a collection of knowledge bases that could be exploited somehow in combination does not provide the right level of granularity.

Another, more sophisticated approach is not to see the Semantic Web as a knowledge base, or as a network of knowledge bases, but as a provider of the base material to constitute a knowledge base of an application/system. The idea here is therefore that a semantic web application is still, in many respects, a knowledge-based system. However, what characterizes the application is the way it creates/forms what can be seen as the equivalent of the knowledge base (i.e., it is a provider of knowledge), by selecting from the semantic web knowledge artefacts of relevance to the task, which together form a workable set of knowledge with useful characteristics (see Figure 2.2).

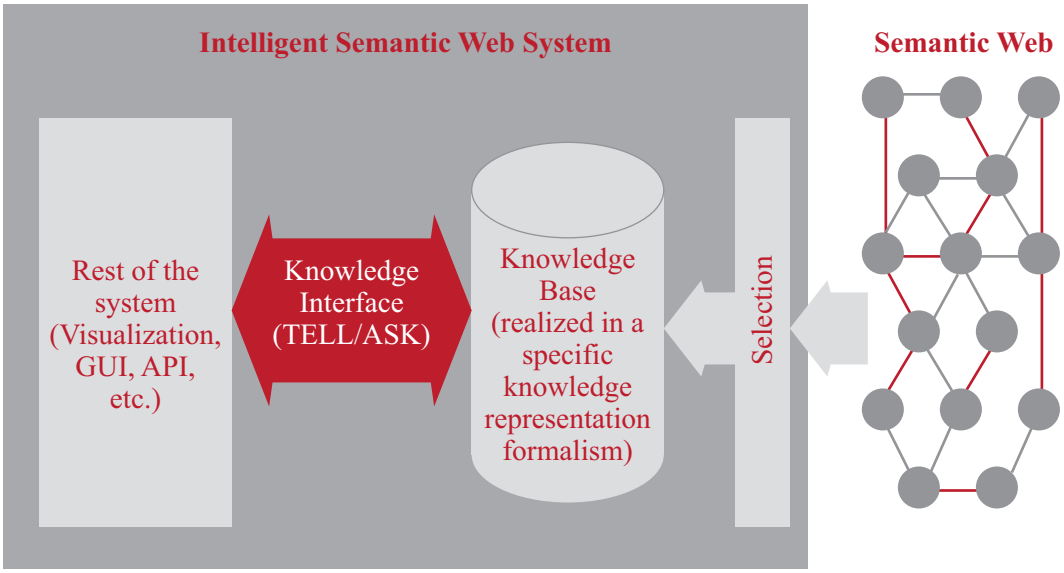

Figure 2.2: The Semantic Web as a provider of knowledge for the knowledge bases of Semantic Web applications.

There are of course many challenges related to the concrete implementation of such an approach. At an abstract level, it also appears unsatisfactory: the proposed approach still considers semantic web applications as knowledge-based systems that include as a component a bounded,

closed knowledge base. In other words, it is disconnecting the application from its primary source of knowledge, semantic web artefacts, to create its own, standalone version of a subset of it. This creates issues from both fundamental and practical points of view. Indeed, concretely extracting knowledge from the Semantic Web and using it in isolation is defeating the purpose of the Semantic Web as a globally distributed knowledge network. It also means that only directly and statically addressable knowledge can be considered, removing some aspects of the flexibility and dynamicity that should be the foundation of intelligent semantic web systems. At a practical level, it raises issues of synchronicity and identity, with the risk of losing the connection between knowledge expressed on the semantic web and knowledge extracted into individual semantic web applications.

As we can see, both proposals above to characterize the way intelligent semantic web systems interact with knowledge from the Semantic Web lead to unsatisfactory results, as they both try to reduce this knowledge to concrete, bounded artefacts on which the characteristics of knowledge bases can be applied [Hendler, 2007]. On the one hand, this appears necessary to allow applications to manipulate and exploit this knowledge, but on the other hand, it fundamentally clashes with the notion of the Semantic Web as a knowledge network, which abstracts from the physical attributes of machines, documents, and data.

Our proposal to address this apparent paradox is to consider the use of semantic web knowledge in semantic web applications similarly to the way described above, but abstracting the knowledge components away from concrete containers that are knowledge bases. Since semantic web applications cannot address the entire semantic web, but at the same time would lose in trying to select and materialise parts of it, they should be characterized by the mechanisms they use to specify their access and manipulation of semantic web knowledge. What we are suggesting here is that an intelligent semantic web system is not a system interacting with a knowledge base drawn from the Semantic Web, or with the Semantic Web as a whole, but an application interacting with a certain *view* over the Semantic Web (see Figure 2.3). "View" is the keyword here, as we consider it both in the sense used in database systems (a way to specify and declare a certain part of the information that can be used to access this part, without separating it from the overall system), and also in the more general sense (a consistent perspective on a broader universe of discourse). In other words, here we aim to combine the best of both worlds, by enabling the system to manipulate knowledge within its own epistemic state, while at the same time this epistemic state defines a formalized connection to the Semantic Web as a knowledge source, rather than as a bounded, self-contained knowledge base.

It is worth noting that this notion of an intelligent semantic web system implementing a view over the semantic web, and being based on other components interacting with this view, is more general than the other aforementioned proposals. The notion of an application using a pre-defined selection of a part of the knowledge expressed on the Semantic Web can still be considered, but this view also allows envisaging applications where knowledge is obtained and manipulated dynamically, without pre-established assumptions on the limits and boundaries of

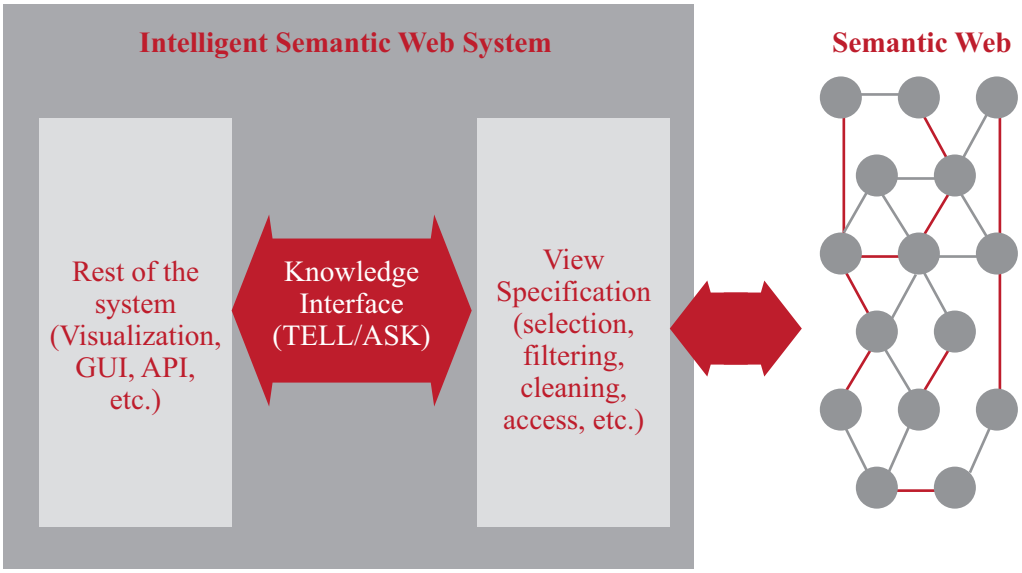

Figure 2.3: Semantic Web applications as views over the semantic web.

what the Semantic Web can provide. In addition this proposal does not imply that the application necessarily materialises the knowledge defined by its Semantic Web view. Here we are focusing, at a conceptual level, on a way to characterize semantic web applications on the basis of the types of views they implement, and the way they implement them.

2.4 INTELLIGENT SEMANTIC WEB SYSTEMS AS VIEWS OVER THE SEMANTIC WEB

Consistently with the analysis above, we consider a semantic web application as a system where one of the components is a mechanism to build a specific view of the semantic web, and communicate this view to the other components for them to realize their tasks (see Figure 2.4).

We believe however that what should characterize intelligent semantic web systems is the way they realize the view specification, which represents their main entry point to the semantic web. Indeed, such view specifications should include the mechanisms to discover, filter, and integrate the knowledge which is to be used by the application, and to which the application might contribute. Also, the view that a semantic web application specifies is, in most cases, left implicit, as it remains embedded in the code of the application and the way it is developed. While this constitutes a challenging issue in the study of semantic web applications, and the development of the semantic web itself [Daga, 2012], it also forces us to think about such views as abstract entities, which might appear with different levels of sophistication and robustness.

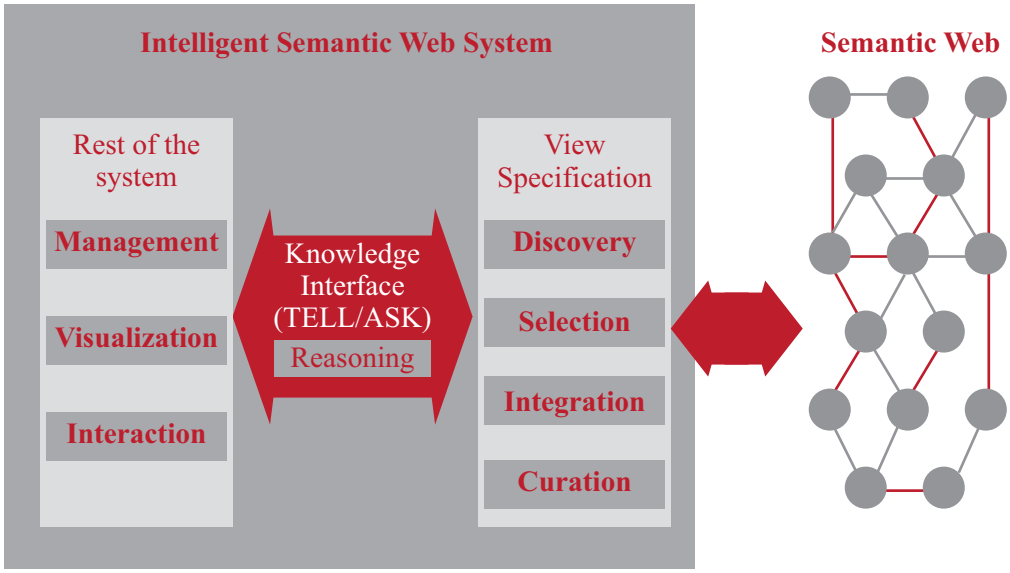

Figure 2.4: Overview of the components of an Intelligent Semantic Web System, characterized as a view over the semantic web.

The basic, static selection of known semantic web resources, constitutes the simplest approach, as it removes some of the challenges described previously related to the manipulation of semantic web knowledge, as it essentially reduces it to something similar to a traditional knowledge-based system, with the difference that instead of using knowledge formalized by experts of the domain, the knowledge base is composed as a snapshot of some web sources. Because it is much simpler than the other approaches, this has been the most common method in the early days of the Semantic Web (see for example Shadbolt et al., 2004), even if it can be argued that much of the novelty associated with being an application of the Semantic Web is lost with such solutions. Also, while this approach is to be considered static, it is still possible for the knowledge view the application creates to be updated, by regularly propagating changes from semantic web sources to its local "knowledge base."

Dynamic selection (i.e., selection at run-time) is the obvious counterpart to static selection, and is of course much more sophisticated. It relies on the idea that the criteria to decide on the inclusion of knowledge from the Semantic Web into the application might be only available at run-time [d'Aquin et al., 2008a]. These requirements might depend on the specifics of the applications. We describe below common categories of selection criteria that might apply. Concrete examples of intelligent semantic web systems which rely on dynamic selection will be described in Chapter 3.

1. **Goal-led**: Where the selection of the relevant information sources is decided on the basis of specific queries or input to the data, depending on the goal of the application. This is especially used in open-domain applications where the possible sources of knowledge for the application cover different domains and are chosen depending on their ability to be relevant to a particular input (e.g., a query). For example, in Chapter 3 we describe the PowerAqua question answering system where the selection of knowledge sources is dependent on the specific question being asked to the system, at run time. The system therefore has to dynamically create the view by finding out what knowledge to include so that the question can be answered.

2. **Provenance/Trust-based**: Where information sources are selected on the basis of their assessed reliability in a specific task of the system. For example, in an application where domain-specific knowledge is required, authoritative sources from the domain might be preferred for critical processes (e.g., domain-specific inferences), while more generic sources can be trusted for simple, non-critical tasks (e.g., named entity recognition). Several examples in the next chapter rely on this strategy, including the highly visible IBM Watson and Google Knowledge Graph, which use sources at various levels of reliability to achieve their task.

3. **Viewpoint-based**: At a higher level of sophistication, we might consider a view mechanism which includes a Semantic Web source on the basis of the viewpoint it expresses, considering in particular its compatibility with the local viewpoint expressed within the application. This is an approach to deal with the inherent inconsistency of the semantic web, and to ensure that the results of the application will be homogeneous and aligned with the intended meaning of the manipulated concepts. This strategy is more subtle and used in a way which is more implicit than the other two. Examples include recommendation systems such as DiscOU, described in the next chapter, which use thematic descriptions of resources that align well with the target query in order to find appropriate resources to recommend.

Naturally, other dimensions might be considered which might be more general or more specific than these three, and there is an obvious need to combine these different approaches to obtain the correct, trusted, and consistent knowledge to support the system's tasks.

2.5 COMPONENTS OF AN INTELLIGENT SEMANTIC WEB SYSTEM

In Figure 2.4, what is included in the "rest of the system" part of the application is not really our focus. Indeed, these are the components which, while not necessarily ad-hoc to the application, are specific to the type of tasks it achieves and the expected interactions it implements with the user. A number of frameworks exist that can support the realization of such interactions, including knowledge/data editors (e.g., Protégé [Gennari et al., 2003], and the NeOn Toolkit [Haase et al.,

2008]), visualization frameworks (see for example Dadzie and Rowe [2011] for a survey in the context of linked data), as well as generic interaction/interface frameworks that are targeted to semantic data (e.g., the Information Workbench [Haase et al., 2011] and linked data browsers such as Tabulator [Berners-Lee et al., 2006]). This part is also in charge of the local information management and storage aspects of the system. Again here, ad-hoc mechanisms can be employed, as well as widely available toolkits and libraries (including a variety of triple-stores [Rohloff et al., 2007] and RDF libraries in many languages: Jena in Java [Carroll et al., 2004], ARC2 or RAP in PHP [Oldakowski et al., 2005], etc.).

Reasoning here has an interesting place. Indeed, if we consider the point of view of knowledge-based systems, as above, reasoning should be an element of the view-management mechanisms, as it is integrated and abstracted "behind the scene," as a logical inference process. However, we consider here reasoning in a broader sense as a process which might generate more knowledge, consistently with the processed view. In this sense, we consider under the umbrella of reasoning something more general than what is typically considered as mathematical, automated reasoning (see for example Robinson and Voronkov, 2001). It relates to the way the knowledge expressed in the view is processed to extract or derive non-trivial results, which might or might not re-integrate the view as newly produced knowledge. It might therefore include formal, logical reasoning as in description logics, as well as other forms of deductive, inductive, or abductive reasoning. It might also be implemented as processes such as data-mining, statistical analysis, or heuristic-based problem solving, which result in new "insight" derived from the manipulated knowledge, while not being explicitly a part of it.

Reasoning is therefore considered "in between" the application side (the "rest of the system") and the view specification side of the system, as it carries out the mechanisms that are required to connect these two aspects. Logic-based reasoning can be achieved through the many existing description logic reasoners that have been designed to work specifically on semantic web technologies (see for example Pellet [Sirin et al., 2007] or Hermit [Shearer et al., 2008]). When employing other forms of data processing as reasoning mechanisms, existing data-mining and statistical engines might be used (for example, combining R [R Development Core Team, 2005] with SPARQL [Prud'Hommeaux and Seaborne, 2008]), as well as mechanisms specifically developed for the purpose of the application (including, for example, information retrieval or collective intelligence mechanisms).

Regarding the interaction between the application side and the view specification side, it can be implemented through a wide variety of ways, depending on the technological choices made in the application. We will in the rest of the book simply assume that there is such an interface in the considered system, even if it might not be explicitly materialised. There is an interesting parallel to make here however between the idea of the TELL/ASK protocol for knowledge-based systems and the protocols associated with the SPARQL specifications. Indeed, the SPARQL update [Gearon et al., 2013] and SPARQL query [Harris et al., 2010] protocols can be seen as the equivalent of TELL and ASK (respectively) for Web (RDF) data endpoints. They encapsulate

in an implementation-independent way (i.e., without specifying how the actual storage of data is achieved, or whether or not inferences are materialised) a generic way of interacting programmatically with the data endpoint. SPARQL however is not specified at the same conceptual level as TELL/ASK, as it focuses on a specific set of technologies and data modeling paradigms. It can be seen as an intermediary layer, used to implement TELL/ASK in a specific context, without the need to focus on the technical components for data management.

We now focus on the specific components used for view specification in Intelligent Semantic Web systems.

2.5.1 DISCOVERY

The first aspect of specifying a view for an intelligent semantic web application is to find out which part of the distributed, heterogeneous, and potentially contradictory knowledge available on the semantic web should be considered. This task can be separated in two steps: discovering knowledge sources of potential relevance, and within these sources, selecting knowledge which is seen as suitable to achieve the task under consideration. The distinctions quickly mentioned above between static and dynamic views, as well as between pre-compiled and run-time views, are very prominent here. Indeed, for many semantic web applications, the set of knowledge sources to employ is manually defined at design time, reducing the discovery component to a list of knowledge sources. An intermediary approach includes applications where a list of candidate sources is pre-compiled within the application, but discovery is still required, to choose which source should be considered in a specific situation. When the potentially useful knowledge sources cannot be predicted *a priori*, especially in open-domain applications, more sophisticated mechanisms might however be required [d'Aquin et al., 2008a]. For this reason, large registries of linked datasets and semantic web sources have been developed, which can be browsed to discover potentially relevant resources.

The linked data community has for example been using the Datahub (`http://datahub.io`), which relies on a CKAN (`http://ckan.org`) repository to collaboratively catalogue and register any datasets published on the Web. The advantage of CKAN is its low barrier of entry. It does not require the datasets to be in any particular format or to comply with any particular access method. Tags and groups are used to provide basic browsing and searching dimensions within the set of datasets, and to gather together data that can be relevant to a specific community or domain—e.g., the open education community with the "Linked Education Cloud" [d'Aquin et al., 2013a]. This however makes it difficult to actually use such resources at run-time for any application, as tagged data sources in CKAN can be very heterogeneous, and implementing the specific data access mechanisms required by each dataset is almost impossible.

More specific to semantic web sources, several initiatives have emerged to build repositories or libraries of semantic resources, especially ontologies [d'Aquin and Noy, 2012b]. These are domain independent—e.g., Cupboard [d'Aquin and Lewen, 2009b, d'Aquin et al., 2009], or alternatively they group ontologies for a specific area of interest—e.g., biomedicine in the case

of BioPortal [Noy et al., 2009] and OBOFoundry [Smith et al., 2007], or eGovernment, in the case of ONKI [Viljanen et al., 2009]. While they vary in the methods they implement to access, browse, and search the knowledge sources they reference, these platforms tend to be more directly usable by semantic web applications, through relying on a restricting set of supported formats for the knowledge sources they reference (e.g., OWL Ontologies).

Moving a step further from collaborative registries of knowledge sources, the discovery part of an intelligent semantic web system can be seen as "finding potentially relevant knowledge sources on the Web in a particular context," making it similar to an information retrieval task. Such a view has been implemented in a number of online systems generally referred to as "Semantic Web Search Engines"—see d'Aquin et al. [2011] for a summary. Earlier versions of these systems tended to focus on ontologies—e.g., Swoogle [Finin et al., 2005], Watson [d'Aquin and Motta, 2011]—while others were built to harness information from the Linked Data Cloud—e.g., SWSE [Harth et al., 2007], Sindice [Tummarello et al., 2007]. In many cases (contrary to standard web search engines), the main focus in building such systems was to provide semantic web applications with extensive programmatic APIs to find, explore, and access knowledge sources from the Semantic Web. For example, as described in d'Aquin and Motta [2011], d'Aquin et al. [2008a], the Watson gateway was specifically designed to support intelligent semantic web systems that relied on the open-domain, dynamic, and run-time discovery of semantic web knowledge resources.

2.5.2 SELECTION

As explained earlier, the idea of defining intelligent semantic web systems as systems relying on the specification of a view over semantic web knowledge is that these systems can then operate upon this view as their own "knowledge base," made of various subsets of formalized knowledge on the web. Once sources of knowledge for such a view have been discovered, a core part of the view specification mechanism of an application is therefore the way in which knowledge is selected to be used/incorporated in the view. Once again, this can vary from a purely static, design-time approach, to a dynamic, run-time approach, the simplest version being the use of discovered knowledge sources as they are, with no selection. Combined with the manual discovery of knowledge sources, this approach however reduces an intelligent semantic web system to a knowledge-based system that simply happens to be using a knowledge base taken from the web.

While the actual selection might not be predefined, another common approach is to rely on a static, predefined specification of the part of the knowledge which is to be used in particular situations. Pragmatically, the way this is implemented in most semantic web applications (especially, linked data mash-ups) is through the application relying on a set of SPARQL queries, or query templates, used to retrieve data from the considered sources. In such linked data mash-ups, this approach is also mainly used in combination with static, manual discovery, since the specification of the knowledge selection process (the query) is generally dependent on the structure and vocabulary employed in the considered sources. Here it is worth mentioning the approach

described in Hartig and Freytag [2012] where the discovery happens as a side effect of selection. Indeed, in this approach, SPARQL queries are processed in such a way that the data sources providing information about different parts of the query are automatically discovered at the time of executing the query.

A more challenging case than the static specification of knowledge selection is the one where the task achieved by the system is open, in such a way that it cannot be predicted in advance what knowledge should be selected, and what will be the structure of such knowledge. This is typically the case of goal-led view specification mechanisms, where both the discovery and the selection of knowledge are dependent on the specific situation and input to the system at a given time—e.g., question answering, as in Lopez et al. [2012] or the interpretation of patterns from data mining in Tiddi et al. [2014a]. In such cases, sophisticated mechanisms need to be implemented that extract from the input given to the system a set of criteria for knowledge selection in the discovered sources—an example of such a mechanism is described in d'Aquin et al. [2007].

Finally, knowledge selection is not necessarily related to choosing the part of the knowledge that will be used to achieve the considered task, but also to filter this knowledge on the basis of some other requirements. As already mentioned, provenance and trust can be used for example to decide whether a particular candidate source is suitable for a specifically critical task. A notion of quality of knowledge might also be used, although this is notoriously a subjective requirement, which can be seen from a variety of different indicators and perspectives, as discussed below in the section on curation. Amongst such quality indicators, one that appears especially important however is consistency, taken in a very broad sense: It appears critical in many scenarios that the view created out of different sources of knowledge does not contradict itself. Logic-based reasoners are therefore often used to test formal notions of consistencies, and other more flexible measures of agreement in knowledge sources have also been defined—see for example d'Aquin [2009a]. In addition, as discussed next, the role of the integration component of the view specification mechanism is often to transform knowledge selected from various sources into a consistently exploitable knowledge set.

2.5.3 INTEGRATION

A central issue of any intelligent semantic web system is to build a view of knowledge available from the Semantic Web that is exploitable for the task at hand. In many cases, such knowledge does not originate from a unique integrated source and requires to be first processed in order to form a consistent and homogeneous base for the application. Again, the most basic systems would rely on static properties of the sources they employ to encode within ad-hoc implementations the mechanisms for reconciling elements of the considered knowledge sources. For the more complex systems however, data and knowledge integration mechanisms have to be put in place, as an intrinsic and often explicit part of the semantic web view specification mechanisms. Accordingly, our approach of characterizing intelligent semantic web systems as view specification fits well

with the classical ways of characterizing a data integration approach, either as *Global as View* or as *Local as View* [Xu and Embley, 2004].

Generally, most semantic web applications would correspond to a Global as View data integration mechanism, with the internal representation model of the system (the view) being a global proxy through which heterogeneous information from various sources can be accessed homogeneously. Achieving this in the context of a complex semantic web system might require the use of (i) sophisticated ontology matching techniques to align the schemas of the various sources to each other or with the application's internal one [Euzenat and Shvaiko, 2007], or (ii) link discovery mechanisms reconciling overlapping entities from different sources—see for example the Silk [Volz et al., 2009b] and Knowfuss [Nikolov et al., 2008] systems. Interesting approaches here include the implementation of ontology matching itself as an intelligent semantic web system, making use of online ontologies [Sabou et al., 2008], the unsupervised configuration of link discovery systems necessary for truly dynamic systems [Nikolov et al., 2012], as well as the combination of link discovery with the discovery of knowledge sources on the semantic web [Nikolov et al., 2011].

2.5.4 CURATION

Knowledge obtained from the Semantic Web can be noisy, inaccurate, incomplete as well as inconsistent. While cleaning (and to a certain extent integration) is about solving these issues, we see this component also in a broader sense: It is about making the obtained formalized knowledge fit for the considered application.

Naturally, the first step to achieve this is to assess this knowledge according to relevant criteria. Data quality evaluation is a difficult problem; simple indicators exist but they rarely capture the requirements of the applications. Indeed, these are often expressed in a way that makes them hard to formalize and test automatically [Lei et al., 2007]. Similarly, the field of ontology evaluation has attracted a lot of attention [Gangemi et al., 2006, Gómez-Pérez, 2004, Vrandečić, 2009] but, for the same reasons as above, it is hard to establish generic assessment and cleaning methods that can be applied automatically at run time. It is therefore often the case that knowledge has to be evaluated *a priori*—e.g., through user evaluation [Lewen and d'Aquin, 2010]; through indirect indicators, such as usage in other contexts [Sabou et al., 2007]; or through an indication of trust in the provenance of the knowledge. Indeed, much current work focuses on making provenance information explicit in semantic web resources [Lebo et al., 2012].

Once this assessment task has been carried out, the resulting knowledge often has to be transformed so that it can be used in the context of the processes in place for the specific system being considered. It might for example be in a different format from the tools used for reasoning, which of course is a trivial matter to solve. At a slightly higher level, it might be at the wrong level of granularity, incomplete, unbalanced, or noisy. In such cases, similar processes to the ones used in data preparation for data mining [Pyle, 1999], such as normalization or aggregation, might be used. From a higher, knowledge level perspective, it may be the case that the knowledge obtained

from the Semantic Web might simply not be modeled in a way that is compatible with the intended processes included in the rest of the system. There are many different ways of modeling knowledge about the same entities, even for dimensions as fundamental as time [Scheuermann et al., 2013]. The application of knowledge refactoring methods [Baumeister et al., 2004, Rieß et al., 2010] might therefore be needed to switch from the modeling approach taken in the built view, to the one required by the system.

2.6 CONCLUSION

We consider the view specification mechanisms as fundamental components that are always found, in various degrees of sophistication, in intelligent semantic web systems. The components described above represent a way to conceptualize the design of such systems, focusing on the aspects that capture the use of semantic web resources, their processing and delivery to the rest of the system. Such a view specification mechanism, while always present, is however rarely made explicit, and the four components are often hidden in the code of the application.

In Chapter 3, we therefore analyze existing intelligent semantic web systems, to illustrate these components and the way they are being implemented in various concrete settings. The goal is to provide an overview of these systems as characterized by their view specification mechanisms—i.e., the way in which they capture a view over knowledge from the Semantic Web to achieve their task. We therefore summarize each of them according to a simple framework relying on the four components described above, providing a basis for comparing both their nature, and the way they are implemented in the different systems. This framework is presented in Table 2.1, which is instantiated for each of the systems described in Chapter 3.

Table 2.1: Framework for analyzing and comparing intelligent semantic web systems

	Type	Implementation
Discovery	Static/dynamic, goal/trust/viewpoint-based	E.g., using specific sources, CKAN, ontology libraries, semantic web search engines
Selection	Static/dynamic, goal/trust/viewpoint-based	E.g., none, query-based, consistency-based
Integration	None, global/local as view, static/dynamic	E.g., data linking, ontology matching
Curation	None, assessment, filtering, preparation, cleaning, reconciliation	E.g., consistency checking, refactoring, query-based filtering

CHAPTER 3

Exemplar Intelligent Semantics Web Systems

In the previous chapters we provided a conceptualization of the Semantic Web and explained how intelligent applications can be built to rely on its contents and knowledge sharing infrastructure. At the same time we also pointed out the specificities of the semantic web, compared to other types of information systems. At this point, we want to shift the focus from the conceptual to the practical, and show examples of concrete semantic web systems.

Of course, despite the model formulated in Chapter 2, deciding on what systems to describe as being semantic web applications is not straightforward. The semantic web, as a research and development area, is following a similar trajectory to the one of Artificial Intelligence: While many may feel that some of the more ambitious initial goals have not yet been met, the techniques and approaches developed within the research community in the semantic web area are being slowly integrated into mainstream systems, such as search engines, in such a way that they might not always be recognized as related to the Semantic Web or to semantic web technologies, even by their developers.

In this chapter, we therefore investigate a range of examples of intelligent semantic web systems, at different stages of development (from large scale commercial applications to research prototypes), which apply techniques and components described in Chapter 2. We chose these applications as they provide a good variety of application types and more importantly: (i) rely on knowledge from the semantic web, following the template of intelligent semantic web systems described in Chapter 2, and (ii) demonstrate novelty in doing so.

Beyond providing concrete, practical examples that can be studied, reused, and reproduced by semantic web developers, as well as concrete illustrations of the conceptualizations discussed in Chapter 2, the goal here is also to obtain an overview of the semantic web application landscape at the time of writing. d'Aquin et al. [2008b], Oren [2008], van Harmelen et al. [2009] provide similar lists of applications to achieve this goal. However, one of the key elements here is that we also wish to contrast concrete developments vs. the conceptual template we have defined in Chapter 2, to better understand what are the remaining gaps, i.e., what seems to be missing from current semantic web applications.

In what follows, we investigate seven applications, which can be loosely categorized into three of the most typical uses of semantic web technologies: (i) search and recommendation, (ii) question answering, and (iii) data analysis and sense making. For each system, we describe

what the system does but also, crucially, why we consider it an intelligent semantic web system. This includes understanding, in accordance to our definitions in Chapter 2, how the system implements a view over semantic web knowledge, as well as the components, technologies, and techniques used that fall into the realm of the semantic web.

3.1 SEARCH AND RECOMMENDATION

It is hard to think of semantic web applications, or even web applications, without thinking about search. The distributed, large scale nature of the Web makes search necessary, and has made web search engine companies powerful entities, which provide a central access point to the vast amounts of information available online. As the Semantic Web is part of the Web, albeit focusing on structured and formalized knowledge, several semantic web search engines have also emerged, addressing the issue of finding and selecting the sources of such knowledge [d'Aquin et al., 2011]. However, as described in Chapter 2, we rather consider these as tools to build intelligent systems that rely on the semantic web, rather than semantic web applications themselves. What we explore here are applications that follow the template for intelligent semantic web systems described previously to make the task of searching the Web easier, more effective, and smarter. We naturally start with the application that, while not being promoted as a semantic web system by its developers, has the most visibility considering this particular scenario: Google Knowledge Graph. We will then describe two systems that achieve one of the search tasks that most obviously benefit from the combination of formalized knowledge, intelligent techniques, and information retrieval: recommendation.

3.1.1 GOOGLE KNOWLEDGE GRAPH

The Google Knowledge Graph, from the title, does not appear to be a system but rather... a graph. Announced in a Google blog post entitled "Things not Strings,"[1] the Google Knowledge Graph does indeed sound very much like the tech giant's own branding and appropriation for the semantic web. Focusing on what it enables however, it appears as a part of Google's search engine that uses semantic web resources and techniques common to intelligent semantic web systems to provide their most visible and successful instantiation.

What the Google Knowledge Graph does in practice is to complement the traditional information retrieval model of Google (i.e., queries are words, which are matched in documents), with an attempt to see keywords as "things" (entities, objects) attached to structured information. This structured information mainly comes from Freebase, a very large, crowdsourced database of more or less everything [Bollacker et al., 2008], which Google acquired in 2010, and later WikiData [Vrandečić and Krötzsch, 2014]. Other sources used by the Google Knowledge Graph include the CIA World Factbook[2] and Wikipedia.[3]

[1] http://googleblog.blogspot.co.uk/2012/05/introducing-knowledge-graph-things-not.html
[2] https://www.cia.gov/library/publications/the-world-factbook/
[3] http://wikipedia.org

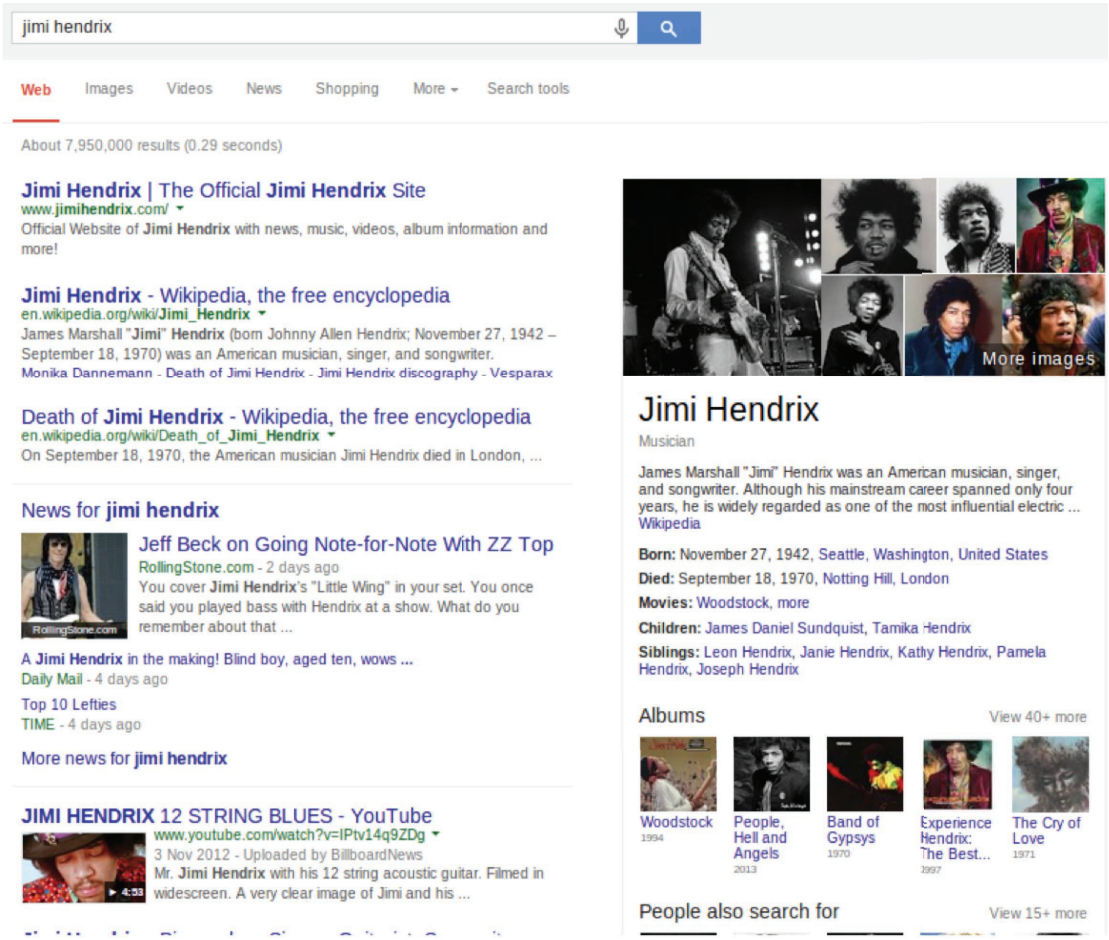

Figure 3.1: Screenshot of a sample result, including "traditional" Google results on the left, and the Google Knowledge Graph panel on the right.

Together with the data, Google also acquired with Freebase some of the techniques that can help in building the "thing-search" facility which represents the most obvious materialisation of the Google Knowledge Graph for the user. In practice what this means is that whenever you search for something that Google can recognize as a known thing—the name of somebody, of a place, the title of a movie, etc.—next to the list of webpages that match these keywords, a panel will appear, giving some basic information about this entity, and links to other related entities (see Figure 3.1).

Table 3.1: Summary of the semantic web view specification of Google Knowledge Graph

	Type	Implementation
Discovery	Static, using pre-established sources	Freebase and WikiData, complemented with other pre-specified sources
Selection	Dynamic, goal and trust-based: related to the query	Likely to be based on a matching of the query string to select candidate entities to be displayed, choosing different sources for different types of entities
Integration	Static global as view	Likely based on reconciling information from used sources with the basic schema of the knowledge graph box displayed in search
Curation	Filtering based on relevance	Likely based on the most common/ popular information to display only the most relevant information

While this might appear relatively simple, there is no doubt that it does match our definition of an intelligent semantic web system. Beyond the trivial aspect that it relies on online, structured, formalized knowledge, i.e., on semantic web resources, it also deploys a set of clever techniques to form a view of these resources that match the need of a specific application and of specific users. While the selection of sources of information is, it seems, rather static (Google only employs a pre-selected set of trusted, known sources), the intelligence of the application comes from the way in which it dynamically, in a goal-oriented fashion, finds what parts of these sources it should bring back to the user. In other words, each result produced by the Google Knowledge Graph defines a very small view over a vast amount of semantic data, which has been carefully, but dynamically extracted to match the user query.

While not all of the details of the view construction process employed by Google are known (see Table 3.1 for a summary), it becomes quickly obvious that this is far from a trivial task, and that it requires both clever algorithms, and a large amount of computing resources. Obviously it is already a challenge to recognize whether or not a set of keywords represents an entity that the Google Knowledge Graph might have information about. Once this has been decided, there might be several different entities matching these keywords. Constructing the view that will be given back to the user therefore must include effective disambiguation methods, to decide, amongst all the possible choices, which one is most likely to be the one of interest. In fact, in many cases, the interface would suggest alternatives, in case the algorithm has made a mistake,

and it is very likely that these choices and other user preferences are being used in the selection process (making the view extraction also a user-oriented one).

Once the entity to be displayed has been selected, the second challenge concerns what information should be displayed. Here again, it is striking to see that this problem is tackled by relying on information which, while not strictly speaking adhering to the most common definition of the semantic web, represents a network of structured, formalized knowledge about the relevant entities, their popularity, common search queries, and the user's own preferences. This knowledge network defines a customized knowledge base for the realization of this task. In other words, the semantic web element here is not only about the way the end-result is presented, but includes also the whole network of semantic information drawn from resources such as Freebase, as well as from Google's own systems.

3.1.2 SEEVL/DBREC

Seevl[4] [Passant, 2011] is a music discovery service, which provides features for artist and music recommendation building on the work on an earlier system, dbrec [Passant, 2010a, Passant and Decker, 2010b]. The idea of dbrec is to use information available in linked data to calculate the similarities that exist between artists and bands based on their semantic distance within the linked data graph. This distance, called the "Linked Data Semantic Distance" (LDSD), is based on the links that connect a particular entity (such as a band or artist) with others of the same type. One of the key advantages of this approach is, naturally, that it uses semantic web resources (the linked data graph) to provide the key information that in other similar systems (e.g., music discovery engines such as Last.fm) would have to be collected and managed by the system itself, thus requiring a "bootstrap" period during which such information needs to be collected, especially for artists and music that are only emerging. The other advantage of this approach is also that, contrary to typical "collective intelligence" approaches which use basic statistical similarity metrics, here the distance between two bands or artists (and therefore the result of the recommendation) can be explained in terms of the links that exist between them. Providing such explanations has evident advantages for the interaction between the system and the user, who can better trust and filter the results. It also makes dbrec itself a semantic web provider, as it makes it possible to provide structured, interpretable results (using the LDSD ontology), which include not only the results of the computation, but also its justification.

As shown in Figure 3.2, mapping the dbrec architecture to the template anatomy of an intelligent semantic web system described in Chapter 2 is fairly straightforward (see Table 3.2 for a summary). Indeed, it appears obvious in this figure that dbrec starts by explicitly building a view of specific semantic web sources (DBpedia, but also Freebase, MusicBrainz, and social media data), by selecting and extracting information about the artists and the music they produce from such sources. This initial, relatively static view (sources are identified *a priori*, even if the selection of the parts to include is partly dynamic) is then transformed, reduced to the relevant information,

[4] http://seevl.fm/

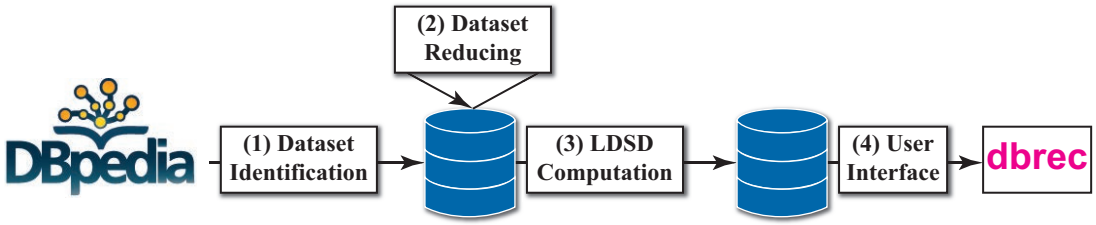

Figure 3.2: Overview of the data workflow of DBRec (from Passant, 2010a).

Table 3.2: Summary of the semantic web view specification for Seevl/dbrec

	Type	Implementation
Discovery	Static, pre-defined sources	Mostly Dbpedia, but also Freebase, MusicBrainz, and social media data
Selection	Dynamics, with static pre-computation goal- and viewpoint-based, depending on the artist/band being considered and the preferences of users	Through the linked data-based similarity/distance metric implemented
Integration	Static global as view, reformulating information about bands and artists to fit the application	Pre-processing of the data about artists and bands and indexing based on similarity and features of a user profile
Curation	Cleaning and preparation	As above, generates a cleaned up version of the data being processed reduced to the features used for recommendation

and refactored for the computation of LDSD. The main task of the system is therefore to exploit this pre-computed, interpreted view of specific semantic web resources on the basis of the specifics of a user's profile or query, and then reason upon it to compute the end-result to be presented to the user.

The intelligence in this system is of course in the recommendation engine that relies on the similarity between bands and artists to find music that might be of interest to a specific user, based on their profile, tastes, etc. However, the core aspect that makes dbrec an intelligent semantic web system is in the preliminary computation it carries out on the semantic web resources it obtained, effectively re-interpreting the knowledge they contain in terms that are of value to the specific applications (what relates them and how these relations might attract interest from the user).

Technically, while much of the core engine of dbrec was developed specifically for this application, the system still uses a set of common components found in many similar semantic web systems, as described in Chapter 2. In particular, it employs a common triple store (Virtuoso[5]) to keep track of the different views it produces over RDF data, a data browsing facility for presentation to the user, and common ontologies for the representation, structuring and sense-making of the processed semantic web information (the already mentioned LDSD ontology, but also the Music Ontology [Raimond et al., 2007]).

Compared to the Google Knowledge Graph, it is much more obvious that dbrec is a Semantic Web system, as it uses technological components based on semantic web technologies and it explicitly promotes itself as such. It is however similarly obvious that it is an intelligent semantic web system, as it relies on a specifically selected and interpreted view over the network of knowledge entities that represent the semantic web.

3.1.3 DISCOU

Similarly, to Seevl, DiscOU[6] [d'Aquin et al., 2012] is a relatively lightweight system that exploits information from the web of linked data to perform a recommendation task. However, while the recommendation approach is relatively simple, the idea is that going through a knowledge graph that represents a machine-readable description of resources makes it possible to recommend things of interest to the user (here, open educational resources) on the basis of other, heterogeneous resources held and managed in a different system, possibly by a different organization (here, TV programs from the BBC).

The motivation behind DiscOU comes from the large amounts of open educational resources available online on many topics and subjects. In particular, The Open University[7] makes such open educational resources available as course material, videos and audio podcasts, etc. All of these resources are described through linked data with metadata available on the Web (within the `data.open.ac.uk` platform, see d'Aquin, 2012a). However, even with such structured descriptions, finding content that is of interest to prospective students is a hard task, as interests might be expressed in various ways. A typical scenario, which is the one DiscOU focuses on, can be expressed as "I watched a TV program on X, and I would like to learn more about it."

What DiscOU therefore does is, first, to analyze all open educational resources that are available, and to index them based on the concepts mentioned in their content. To achieve this, it uses DBpedia Spotlight [Mendes et al., 2011], a named entity recognition system that returns, from a piece of text, URIs of entities in DBpedia mentioned in the text. Each resource is indexed with the entities found, with a weight corresponding to the relevance score given by DBpedia Spotlight. To then recommend such resources from a BBC program, the synopsis of the program is first obtained from the linked data description of the program on the BBC website, and this

[5] `http://virtuoso.openlinksw.com/`
[6] `http://dicou.info`
[7] `http://www.open.ac.uk`

synopsis is put through the same process of named entity recognition. The program can then be matched to a set of resources in the index, which are returned, ranked, to the user.

In practice, DiscOU takes the form of a bookmarklet that can be triggered when looking at a BBC program page or an iPlayer page.[8] When triggered, DiscOU displays a "recommendation box" on top of the page, showing the 10 most relevant open educational resources from the Open University (see Figure 3.3).

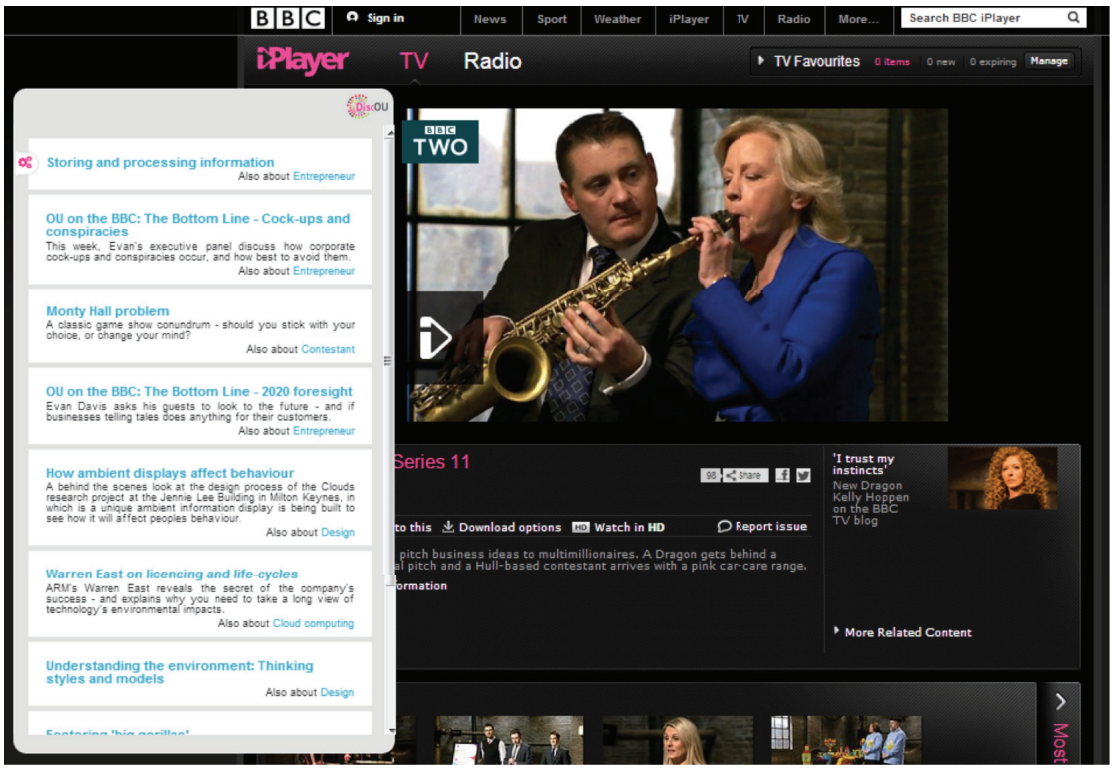

Figure 3.3: Screenshot of the DiscOU interface on an iPlayer page.

A key benefit from relying on knowledge from the Web (information about the programs, educational resources, and the entities they relate to) is that it enables DiscOU to connect resources of different types across heterogeneous systems, effectively using DBpedia as a bridge between BBC programs and Open University content. Like in Seevl, it also makes the recommendation process more transparent. For example, the interface displays next to each recommendation an indication that the recommended resource is "Also about X," with X being one of the topics it shares with the BBC program being looked at. This is achieved by looking at the overlap

[8] iPlayer is the video player of the BBC, see http://www.bbc.co.uk/iplayer.

between the DBpedia entities in the two resources, at their computed relevance scores for each of the resources, and also at their connections within the linked data graph, which help the system to figure out the most prominent one.

This transparency also has another direct advantage: better user control. Indeed, since recommendation is based on a knowledge-based representation of the content of the resources, it is possible to edit this representation to obtain matching resources that are more directly of interest to the user. This is illustrated in Figure 3.4, showing how the weights of the topics associated with BBC programs can be changed by the user, for their own session, to reflect better their own interest and trigger a new set of recommendations.

While relatively simple and lightweight, DiscOU is an interesting example as it illustrates many aspects of the Semantic Web (use of linked data, knowledge from the web, and smart knowledge processing mechanisms) and of intelligent semantic web systems (see Table 3.3). It implements a static, pre-selected view as the index of the semantic web in the form of the index of open educational resources, but also a dynamic view based on bringing into the system, at runtime, knowledge about the BBC program being considered and about the entities that relate to it. In the terminology of knowledge-based systems, DiscOU has a different epistemic state for each user, created depending on the BBC program they are watching, which evolves every time the user changes the weights of the topics associated with the program. To construct this dynamic view, DiscOU employs various sources from the semantic web (Dbpedia, BBC, and `data.open.ac.uk`), and reasoning mechanisms (named entity recognition, similarity search, overlap analysis). The combination of these components provides DiscOU with all the elements required for intelligent recommendation.

3.2 QUESTION ANSWERING

The search and recommendation systems described above fall into the general category of information retrieval systems. Information retrieval defines one of the most common research areas on the web, and also one of the biggest challenges for researchers in the semantic web area. Typically, it relies on a base of information (often encapsulated in documents) from which a part can be extracted that matches a specific information need from the users. In the systems above, the information need is not expressed directly, but implicitly in the user query ("I want information related to these concepts, expressed in keywords" or "I want information that relates to my interests, which can be seen through the use of other resources"). In these cases, most of the intelligence comes from the interpretation of such a query as a representation of an information request from the user.

In this section, we look at systems that go a step further by making this information need explicit: Question Answering Systems. Question answering systems are tools that take an explicit information request from the user—a question such as "What is going to be the weather tomorrow?," and match it to the information base containing potential answers. Interestingly, while the information need is here explicitly expressed by the users, many of these systems, especially

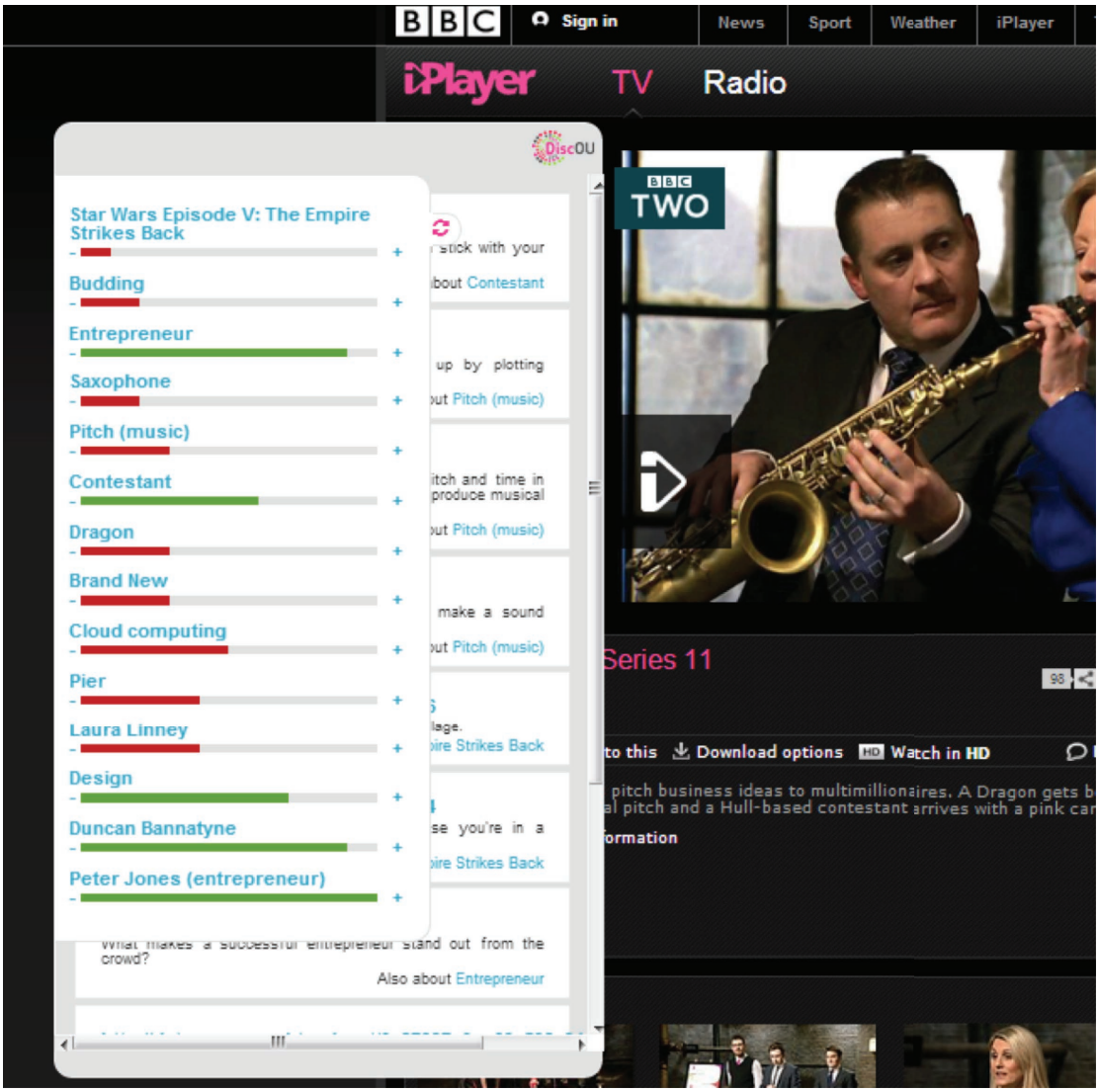

Figure 3.4: DiscOU's personalization interface.

when taking questions in natural language, still put a strong emphasis on the interpretation of the question. Moreover, with more precise questions comes the expectation of more precise answers. As intelligent semantic web systems, these tools are therefore interesting for the reason that they can exploit the Semantic Web as a vast source of potential answers, while having to deal

Table 3.3: Summary of the semantic web view specification for DiscOU

	Type	Implementation
Discovery	Static, based on pre-defined sources	Open University content (data.open.ac.uk), BBC program, and DBpedia
Selection	Dynamic, goal and viewpoint-based, depend on the selected resource	Use DBpedia Spotlight to select DBpedia entities that connect the selected resources to indexed ones, complemented by user-defined view of the description of the selected program
Integration	Partly static, partly dynamic global as view	Pre-computed index of open educational resources based on relevant DBpedia entities, runtime computation of comparable representations of BBC program
Curation	Simple, static filtering of displayed information	Information presented to the user is filtered to only contain the most relevant aspects, relation between recommended resources and the query resource are summarized into one topic

with the added challenges (mentioned in Chapter 1) of having to process such a distributed and heterogeneous source of knowledge to turn it into a proper answer-base.

As with the previous category, we start by describing a system which, while not promoting itself much as a semantic web system, has been developed within an industrial/commercial context (as opposed to the second system which is purely academic) and has become extremely famous: IBM Watson.

3.2.1 IBM WATSON

It would be difficult to find a more representative and famous example of a modern "intelligent system" than IBM Watson [Ferrucci, 2012]. Indeed, Watson is an Artificial Intelligence system, created by the DeepQA project in IBM, built to provide accurate answers to questions posed in natural language (English) on more or less any topic, from politics to movies and entertainment. Most famously, Watson was designed to be able to compete in the famous TV game Jeopardy, and actually went on to beat the best human players in the world [Chu-Carroll et al., 2012].

To find and validate answers to the questions asked during the game, Watson relies on a variety of different data sources, both structured and unstructured, including for example the full

text of Wikipedia. As shown in Figure 3.5, it implements a complex workflow (i) to process these sources to make them usable for the purpose of either generating candidate answers or validating these by providing evidence; (ii) to process the question to transform it into something that can be matched against possible answers; and (iii) to synthesize the answers and associated evidence to select the most likely one and formulate it in the appropriate way.

While it is obvious that IBM Watson fits the definition of an intelligent system [Ferrucci et al., 2010], it does not directly follow that it is an intelligent semantic web system. Indeed, it is important to realize (both for the purpose of analyzing the system and for understanding the settings in which Watson was competing in the game of Jeopardy) that during the game, Watson was not actually connected to the internet. All sources of information were imported and accessed locally in one (rather large) piece of dedicated hardware.

From a purely technical point of view however, it is already clear that Watson is making use of semantic web resources, including DBpedia, which is a linked data version of Wikipedia. More conceptually, it is easy to map the architecture of Figure 3.5 to the template intelligent information system presented in Chapter 2 (see Table 3.4). Structured and unstructured data from the Web are first extracted into a very large static view—a very large knowledge base of more or less anything, from which information relevant to the question is dynamically identified. Results are then provided to the module in charge of selecting the answer with the best confidence score and presenting it to the users.

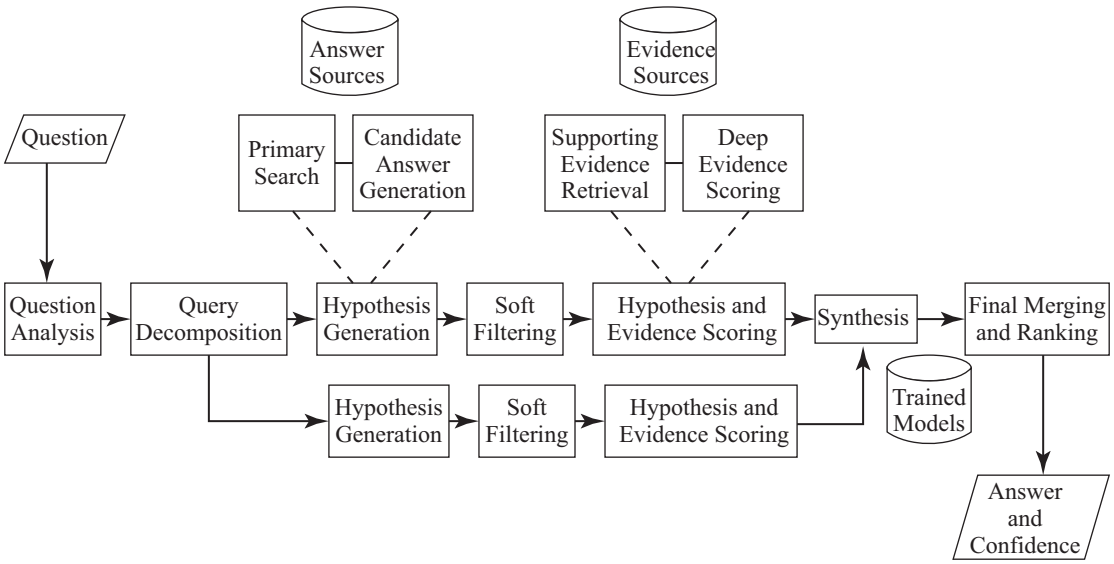

Figure 3.5: Overview of the workflow of Watson (from Ferrucci et al., 2010).

Table 3.4: Summary the semantic web view specification for IBM Watson

	Type	Implementation
Discovery	Static, based on pre-defined sources	Large number of sources, including Wikipedia pages
Selection	Dynamic goal and trust-based, dependent on the query and confidence in the source	Different sources of information are used to find different kind of evidence for an answer, and are considered according to confidence in the strength/accuracy of the evidence
Integration	Dynamic, local as view	Evidence collected is aggregated in support of specific answers
Curation	Static data preparation and dynamic filtering	All sources compiled in a large knowledge base, and evidences filtered at run-time based on confidence and relevance to the question

3.2.2 POWERAQUA

While less famous than IBM Watson, PowerAqua [Lopez et al., 2012] aims at a similar goal: finding the answers to natural language questions in large amounts of heterogeneous semantic web resources. There have been several different versions of the PowerAqua system, starting with the AquaLog ontology-based question answering tool [Lopez and Motta, 2004], to the linked data-based version of PowerAqua [Lopez et al., 2010]. The basic functioning of PowerAqua is that it takes as input a question in natural language (e.g., "who are the members of the rock band Nirvana?"), and produces an answer, which can come from many different sources, and can actually combine partial answers from these resources (see Figure 3.6).

To achieve this functionality PowerAqua starts by analyzing the question using the Gate natural language processing library [Cunningham, 2002], to transform it into a set of "linguistic triples." These triples extract elements of the question (type, subject, relations, etc.) and represent them in patterns similar to graph matching patterns (e.g., "?x Member Nirvana" and "Nirvana typeOf RockBand"). Simplifying what is a very sophisticated process, these triples are then matched to the content of semantic web sources, i.e., to actual triples in RDF, by first matching their elements to existing ones in these sources and then reconstructing the structure of the triples as it finds clues to the presence of an answer.

What makes PowerAqua an obvious intelligent semantic web system and a different tool from other question answering systems, including IBM Watson, is in this last step, and especially in the way the sources are matched. Indeed, PowerAqua is built so that it can dynamically search

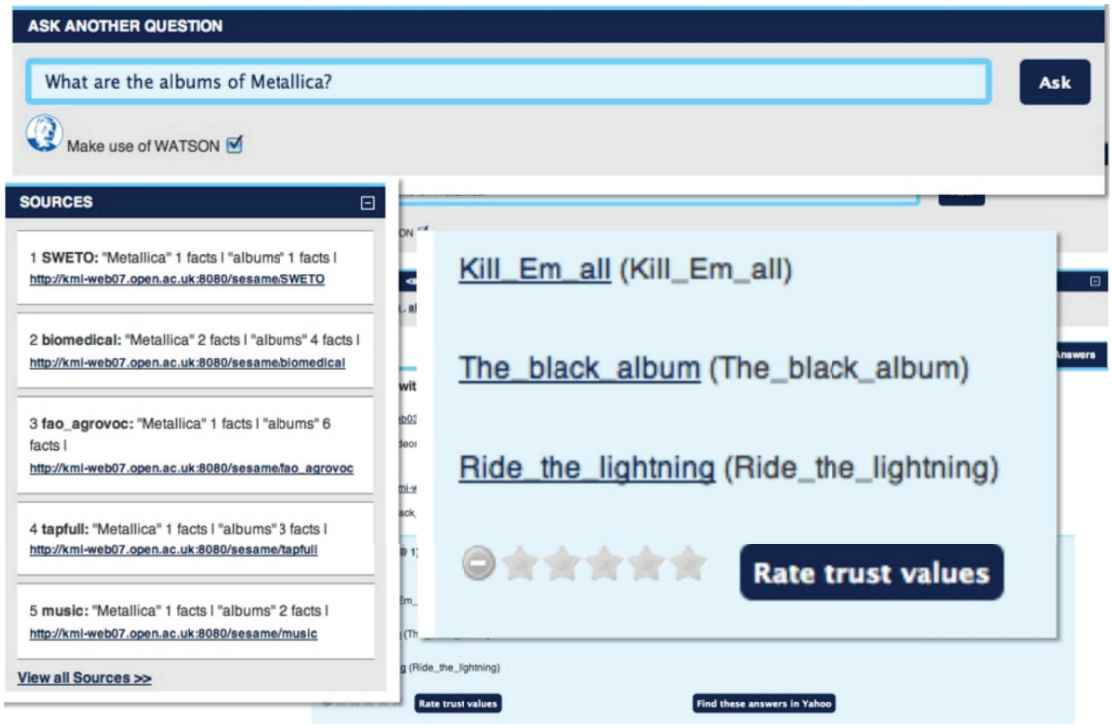

Figure 3.6: Screenshot of PowerAqua with the question "What are the albums of Metallica."

for sources which might contain answers to the question (in the form of matching components to the constructed linguistic triples). It can use a combination of multiple local repositories of ontologies/semantic data and semantic web search engines to find knowledge sources with candidate matches. In other words, PowerAqua's view on the semantic web, its epistemic state, is entirely dynamic and goal oriented. Starting from the question, it builds, at run-time, a knowledge base dedicated to answering this question, selecting relevant sources and, within these sources, knowledge that can support the question answering process. PowerAqua then reasons upon these sources and their combination, to finally produce an answer for the user.

So, in contrast with pretty much any other QA system which has been proposed over the years, PowerAqua does not assume that all the knowledge has been pre-compiled and homogenized in advance, but it is able to deal with heterogeneity at run-time, using a variety of heuristics to decide whether different ontological representations of a domain may be connected at run-time. This particular way of specifying a semantic web view is summarized in Table 3.5.

Table 3.5: Summary of the semantic web view specification for PowerAqua

	Type	Implementation
Discovery	Dynamic, based the elements of the question	Uses a local repository and semantic web search engines to find sources able to answer the question
Selection	Dynamic, goal-based	Match the elements of the question, represented as "query triples" to the assertions made in the discovered sources
Integration	Dynamics, local as view	Aggregate elements of answers from different sources into different answers, using ontology matching techniques to reconcile different representations of the same entities
Curation	Dynamic filtering and ranking based on confidence	Uses the confidence of the matching between the question and the answer to rank results

3.3 DATA ANALYSIS AND SENSE-MAKING

The two previous types of intelligent systems can be seen as intelligent ways to find and process existing knowledge and information. However, one of the key points of semantically representing such information on the Web is to enable the derivation of new knowledge. In this sense, data analysis and sense-making can be seen as the key reasoning tasks for the semantic web: extracting new insights and understandings from the large amounts of information available directly from the web. Once again, we show two illustrative examples, one with a commercial background and one with an academic background, which achieve this task using semantic web principles and technologies.

3.3.1 GARLIK DATA PATROL

Garlik Data Patrol [Harris et al., 2010] is an online service that checks, on behalf of the user, information about them that appears online, and in various (official or not) information sources. The idea is that such a monitoring service can help users in detecting how information about themselves appear online that could be used for, or represent early signs of identity theft. The system first collects from the user information that needs to be monitored and will alert them

every time such information has appeared in other information sources in a way that appears to be suspicious.

Garlik Data Patrol has been described as one of the first real, commercial semantic web-based system. It relies on the aggregation of information from many different sources, as well as the processing of this information and of knowledge about their sources (see Figure 3.7). As an early system relying on semantic web technologies, a lot of discussion around Garlik Data Patrol focused on the technical aspects of the development of the system, and especially on the benefits of the use of these technologies (see Harris et al., 2010).

Nevertheless, Garlik Data Patrol is also a very interesting tool from the point of view of intelligent semantic web systems, beyond its information scale and its sophisticated technical architecture (see Table 3.6). On the one hand, it can be seen as a typical data-based system, aggregating data into a central repository. However, looking at it as an intelligent semantic web system, this aggregation is already a complex task. Indeed, while not all details are available, it is clear that this requires selecting the sources and processing them to obtain the right level of information, as well as including information about each source and the information they contain. This selection is likely to be partly static, but certainly requires an element of dynamicity as new resources might emerge that contain information of relevance that needs to be integrated in the view (possibly very rapidly, considering the potential criticality of the information). This constantly evolving view, drawing from streams of data from various sources, is then processed by specific reasoning mechanisms in order to detect potential threats to the users of the service.

Table 3.6: Summary of the semantic web view specification for Garlik Data Patrol

	Type	Implementation
Discovery	Mostly static, with update based on crawled sources	Uses various known sources and web crawling
Selection	Mostly static, based on information from the user	Querying collected sources for known information
Integration	Static integration of dynamically crawled sources, global as view	Newly crawled sources integrated into global knowledge base, and indexed based on available user information
Curation	Processing, indexing, and transformation of sources	Sources are processed to extract user information and transformed into a common representation

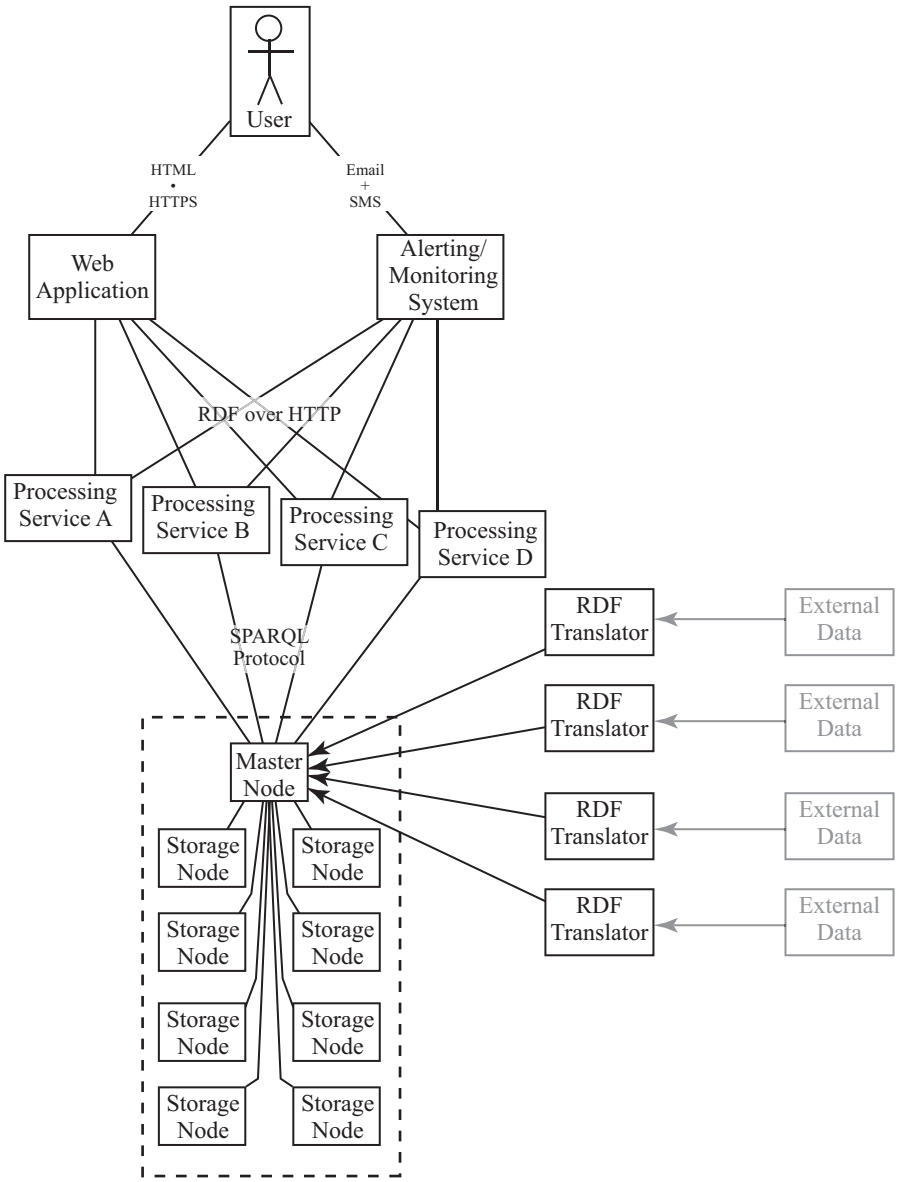

Figure 3.7: Garlik data patrol architecture (from Harris et al., 2010).

3.3.2 REXPLORE

In a different domain, Rexplore [Osborne et al., 2013] applies a similar sense-making approach to Garlik Data Patrol and therefore can be considered as an intelligent semantic web system, which carries out the integration and intelligent analysis of large amounts of data from several different sources. Rexplore is a tool which allows users to explore and make sense of data associated with research activities, making it possible, for example, to investigate which trends are emerging in a particular research area and how research communities merge, split, emerge, or cease to exist. In addition Rexplore makes it possible to explore the research trajectory of individual researchers (see Figure 3.8), as well as performing expert search using a variety of parameters.

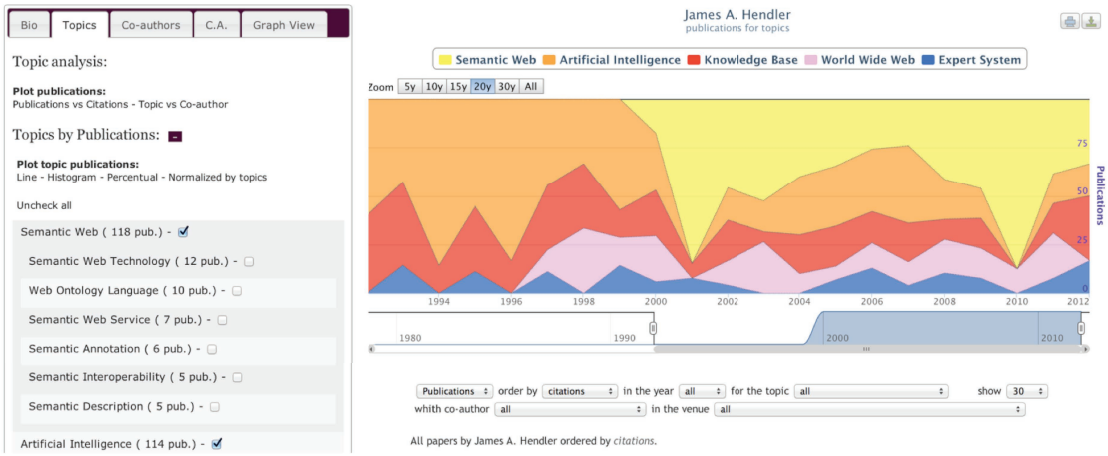

Figure 3.8: Screenshot of Rexplore, looking at the trajectory of a researcher (James A. Hendler) in several research areas.

Rexplore distinguishes itself from the many other tools providing research analytics and bibliometrics not only through the variety of analyses it can provide, but also through the semantic characterisation of the research areas and communities it relies on. It does not rely on a pre-established set of areas, but on a specialized ontology generation algorithm (see Osborne and Motta, 2012), which identifies research areas at various levels of abstraction from large corpora of publications and then creates a fine-grained semantic network that relates these research areas using hierarchical relations, such as "X is a sub-topic of Y." Having constructed this fine-grained topic structure, it is then able to associate each publication to one or more topics, and therefore also link these to authors, organizations, countries, publication venues, etc. To build this broader knowledge base, Rexplore integrates data from a variety of sources, including DBpedia, geonames, repositories of calls for papers, and others, in addition to integrating data from a variety of bibliographic repositories. An interesting aspect here is that the resulting knowledge base is highly dynamic, as the space of research data is constantly evolving.

The sophisticated sense-making and data analysis abilities of Rexplore, based on the combination of several heterogeneous knowledge and information sources, make it a prime example of an intelligent semantic web system, with an emphasis on "intelligent." The views it builds (see Table 3.7) are based on sources that are identified *a priori*, and with a pre-established selection process to extract the relevant pieces of information within them. It is however not only a static view, as it includes the evolution of this information as a critical aspect. It also employs a combination of reasoning processes, such as the advanced topic detection algorithm, and semantic clustering techniques which rely on semantic similarity measures to identify "virtual" communities of researchers, who appear to follow similar research trajectories—e.g., researchers who were primarily active in AI in the 90s but then switched their focus to semantic web research in the following decade.

Table 3.7: Summary of the semantic web view specification for Rexplore

	Type	Implementation
Discovery	Static, based on pre-defined sources	Uses source of bibliographic data, as well as general information sources such as DBpedia and geonames
Selection	Statically defined, updated based on evolution of base sources	Extract information about the topics, keywords, and basic metadata of research articles and researchers to compute analytics
Integration	Static, global as view	Integrate selected information into a global base, which is queried at runtime to obtain information about researchers and articles
Curation	Cleaning and re-indexing of source information based on extracted topics	Information about articles is used to extract a taxonomy of research topics, forming the base dimension for indexing in the application

3.4 CONCLUSION

Interestingly, the three categories of intelligent semantic web systems described and illustrated above are not very far from the natural expectations for the primary application areas for the semantic web. Finding and querying information is certainly the most common use case of the web, and making sense of such information is what can be expected as a result of the semantic characterization of such information. It is however interesting to see that the reasoning mecha-

nisms and the way in which knowledge from the Web is selected and processed has turned out, in these examples, to be different (and in many cases less sophisticated) than one might have expected when the vision of the Semantic Web was first expressed [Berners-Lee et al., 2001]. Indeed, for example, only a few of the systems described above use the logic-based reasoning mechanisms that are traditionally associated with knowledge-based systems, and when they do, they do not constitute the primary inference mechanism at the basis of the system. This might be due to the complexity of using such reasoning mechanisms on information that is not only of larger scale, but also faces much more heterogeneity and noise than what is typically associated with closed, knowledge-based systems. It might also simply be the case that such mechanisms are less appropriate to the tasks that are relevant to intelligent semantic web systems, than they are to expert systems. Whichever the case, it is clear that both aspects will need to be further developed. Indeed, as can be seen from the previous examples, there is still scope for intelligent semantic web systems to become more effective in processing open, distributed information from the Web. While the previous chapters have shown that the idea of using the entire Semantic Web as a knowledge base is not what characterizes intelligent semantic web systems, many of the current applications still resemble closed knowledge- (or even data-) based systems, with statically selected and integrated information sources being processed with relatively simple mechanisms. In other words, this might only be the beginning of the development of intelligent semantic web systems, as much of the benefits shown in the examples above are still embryonic, and several obstacles stand in the way of the full adoption of the intelligent systems approach to development. The next chapter concludes this book by focusing on these obstacles.

CHAPTER 4

The Challenges that Need to be Addressed to Realize Ubiquitous Intelligent Semantic Web Systems

As shown in the previous chapter, examples of intelligent semantic web systems are becoming more and more common, and have evolved from academic prototypes demonstrating advanced concepts, to being adopted by commercial organizations producing tools used on a daily basis by millions of people. Hence, the vision of the Semantic Web and of what could be done with it [d'Aquin and Motta, 2011] has gone a long way: semantic web technologies are now robust enough to be adopted commercially and integrated in mainstream tools and tasks, without users being necessarily aware that the deployed solutions originate from semantic web research. While we can be happy that this has been achieved in what, even in the technology area, can be considered as a short period of time, many of the promises of intelligent semantic web systems are still not fulfilled: several challenges have yet to be overcome.

Many of these challenges are, naturally, related to the technology—after all, with all their successes, intelligent semantic web systems still represent a paradigm shift with respect to the way information has traditionally been handled. However, as the technology evolves, other factors are also impacting on the further development of intelligent semantic web systems. These are the factors that relate to the fact that, even with the Semantic Web being dedicated to machine processing of information, humans are still involved, who need to interact with the information being processed, to deal with its diversity, and to make sense of the complex network of rights that might be associated with such information.

4.1 TECHNOLOGICAL CHALLENGES: SCALE, ROBUSTNESS, AND DISTRIBUTION

In Chapter 2, we argued that Intelligent Semantic Web Systems can be seen to have evolved from Knowledge-based Systems, finding their origins within the now well established Artificial Intelligence, Knowledge Representation, and Knowledge Engineering fields. From this perspective, the challenges associated with the development and further adoption of such systems strongly relate

to the changes that the environment of the Semantic Web introduces. As described in d'Aquin et al. [2008a], these changes mostly relate to four different factors:

1. *Heterogeneity*: In the world of Knowledge-based Systems, tools and applications are typically built from a limited set of sources of knowledge/information, while intelligent semantic web systems assume knowledge comes from a variety of sources, which may differ in modeling, scope, interest, etc.

2. *Quality*: A side effect of building systems in a closed environment with a limited number of sources is that the quality of the knowledge/information fed into the system can be controlled, which is of course not the case when using many external sources to build a view which we cannot assume to control.

3. *Scale*: While any Intelligent Semantic Web System might not require to build on views including a very large amount of knowledge and data, the available resources are already much larger than typical closed, purpose-built knowledge bases.

4. *Reasoning*: Traditional knowledge-based systems employ complex, sophisticated reasoning procedures, which are made feasible by the fact that the knowledge on which they relied was restricted, in size and variety. As shown in Chapter 3, many Intelligent Semantic Web Systems rather employ mechanisms that exploit the larger scale and greater richness of the information they can obtain from the Semantic Web.

These factors introduce challenges that have made Semantic Web research move toward integrating approaches from several other fields in recent years. Scale in particular has attracted a lot of attention, leading many researchers to turn to the area of very large databases. Technologies such as SPARQL and triple stores have therefore evolved from being able to handle medium size databases to becoming high-performance engines capable of holding and processing billions of RDF triples [Morsey et al., 2011].

The main issue of scale, as mentioned above, has however not been addressed. Indeed, by focusing on making triple stores better and more scalable, which is naturally a good thing, what has been created is to a large extent a new form of database systems, capable of storing and querying large amounts of locally stored and locally processed data. In the scheme of Chapter 2, this is naturally very beneficial to the development of intelligent semantic web systems, as it enables them to build scalable and usable data management infrastructures that can naturally and with low effort integrate information from the semantic web. Also, of course, SPARQL is designed for Web Data and many of these triple stores include basic deductive reasoning mechanisms that can, to an extent, exploit the semantics of the representation language and ontologies. By going in this direction however, we take the risk of building a web of a few very large databases rather than a truly distributed semantic web.

Another interesting aspect emerges when starting to address the technical challenges of the semantic web, especially scale, using database approaches: robustness. Indeed, much like Intelli-

gent Semantic Web Systems differ from Knowledge-based Systems by their openness to multiple, external sources of knowledge, the Semantic Web differs from databases by the openness with which they are being made available. Indeed, semantic web resources available through tools, such as SPARQL, can be queried over the web, with in most cases very few restrictions into what can be queried, or how many computing resources can be used to answer a query.

This certainly makes sense! If intelligent semantic web systems are to be built using these resources combined with others, it would be a step backward to restrict what information and resources they could access to a limited set of scenarios specified by the provider of the information. Some proposals exist in this direction, especially around "hiding" such query endpoints by means of pre-defined APIs.[1] However, in the case of many of the intelligent semantic web systems we have considered in the previous chapter, this approach would limit their ability to achieve tasks that were not envisaged by the information provider without significant additional development effort.

We are therefore facing a complex and potentially blocking situation: intelligent semantic web systems require open, unrestricted access to information through endpoints providing sophisticated query and data processing mechanisms. This certainly represents a shift from the traditional environment in which very large databases are being developed, where complex querying and processing is only available to the administrator of the database, and the ability for external agents to access the content of the database is restricted to a few pre-defined and well tested channels. It is also a step away from the architecture of the Web, which, through distribution and relying on a very simple and "cheap" (in terms of computation) mechanism for information access, has robustness built in. The challenge is even more critical as, with the increasing number of intelligent semantic web systems, or at least of systems relying on semantic web resources, requirements for high availability are also growing. Proper industrial adoption of such resources is very much dependent on addressing this challenge, beyond examples such as Google and IBM Watson which can afford to deploy their own large infrastructure to "replicate" the required semantic web resources. Studies have started to show the extent of the problem [Vandenbussche et al., 2013] and a few proposals exist for specific technological solutions to improve the availability, especially of SPARQL endpoints [Verborgh et al., 2014]. These solutions however appear more like patches to a technology stack borrowed from databases and simply "put on the Web." In other words, research is needed on the properties of intelligent semantic web systems and of semantic web resources to understand the way in which not only specific technologies such as SPARQL, but also the more fundamental, global architecture of the Semantic Web can enable robust interactions between them (see for example Hartig et al., 2009). Interestingly, this might lead to re-considering the contribution of fields which have been left out of semantic web research in the last few years, such as agent-based systems and peer-to-peer systems [Hendler, 2001, Staab and Stuckenschmidt, 2006].

[1] See for example `https://github.com/UKGovLD/linked-data-api` and Daga et al. [2015c].

Interestingly, this discussion leads again to one of the most fundamental notions at the basis of the semantic web, which the technology is still struggling to fully exploit: distribution. As mentioned above, distribution should be the prime mechanism by which scale and robustness are achieved, and is in principle one of the pillars of the architecture of the semantic web, since this relies in theory on the architecture of the web. In practice however, distribution remains an issue that is not easily addressed when applied to so many different levels of a system, from information storage and exchange, to reasoning and interaction. Indeed, as already pointed out, SPARQL does not actually rely on the architecture of the Web, as it is currently mostly a protocol and query language to create data endpoints on the web. Again, proposals exist to address the issue, starting with the notion of SPARQL federation [Prud'Hommeaux and Seaborne, 2010] and many research initiatives are currently ongoing to turn it into an effective solution (let alone a scalable and robust one; for example Buil-Aranda et al., 2011, Schwarte et al., 2011). As a result of still having to deal with such low level challenges to achieve what seems to be one of the most crucial properties of the Semantic Web (the ability to seamlessly deal with information from various sources distributed over the Web), intelligent systems that can truly exploit the idea of traversing a "global data space" [Heath and Bizer, 2011] remain rare (see Tiddi et al., 2014a for an example), and have to face many other challenges not yet addressed by the semantic web stack, such as knowledge discoverability [d'Aquin et al., 2011], quality, and heterogeneity (see for example, Tiddi et al., 2014b).

An interesting observation is that some of the solutions to these technological problems might come from re-connecting the semantic web with some of its roots in Artificial Intelligence, and especially knowledge-based systems, where notions of meta-reasoning, uncertain search and decentralized reasoning (see for example d'Aquin et al., 2013b) are well established and might find a new larger scale application. It is also interesting to note that while technological solutions have been the main focus of research for these issues, many of them relate to more conceptual challenges, as discussed in the following section.

To summarize:

1. The scale issue associated with intelligent semantic web systems might not be so much about dealing with a few very large information sources, but more about dealing with a very large number of (not necessarily big) information sources.
2. This emphasizes the need for better solutions to ensure the robustness and reliability of semantic web resources, as the (even temporary) inaccessibility of one resource might jeopardize the use of a whole network of connected systems and other resources.
3. This issue of robustness is especially prominent since, by making information resources open for reuse, the Semantic Web also makes them more susceptible to breakage.
4. It also emphasizes the need for further research on the fundamental notion of distribution (of information, knowledge, and their processing). Distribution is at the essence of the

semantic web, even if it is (as described in Chapter 1) hidden behind its conceptual structure. Because of such uncontrolled distribution, it is not sufficient to look at database systems for potential solutions to the robustness and scale challenges described above.

4.2 NON-TECHNOLOGICAL CHALLENGES: THE HUMAN IN THE LOOP

Most of the technological problems discussed above relate to some form of scalability, which is not necessarily related to size. In fact, size is probably the easiest dimension of scalability, while the most difficult aspect of scalability is not to deal with large volumes of the same kind of data, but to deal with volumes of many different kinds of data. Semantic web technologies have very much focused on one way of being "of many different kinds," data and knowledge heterogeneity [Suárez-Figueroa et al., 2012]. The Big Data community typically talks about variety [Laney, 2001], which sometimes has the same meaning as heterogeneity, but is often used with a much broader meaning. Somehow, there is one aspect of scalability which is much harder to address by means of purely technical means. In d'Aquin et al. [2014a] we called it *diversity*: the fact that data and knowledge not only come in different formats and subscribe to different modeling principles, but also that they originate from different sources, might be of different scope and quality, and might be distributed under different constraints, with different regulations applying to them, etc. This is very much a scalability issue, as we cannot expect to be able, as the semantic web promises, to automatically process large amounts of knowledge from a large number of sources, if for example each source needs to be manually inspected to figure out whether or not we have the right to use it. Diversity here relates to the human part of data processing, where decisions need to be made on which data/knowledge sources to use and how. If we want intelligent semantic web systems to work and grow, we need to minimize the human effort required to make such decisions. We need to make the technology scale.

4.2.1 QUALITY

Amongst the most notoriously difficult forms of diversity in knowledge and data is of course quality. Much like the sentence "you can't trust stuff on the internet" has become common sense when talking about information on the Web, trust in data and knowledge from the semantic web will undoubtedly become key to the expansion of intelligent semantic web systems. In particular, given the template of Intelligent Semantic Web Systems defined in Chapter 2, it is crucial that these systems are able to automatically select and exploit knowledge and in order to do this they need to be able to reason about trust. As shown in Chapter 3, this is one of the most complicated tasks that only a few of the simpler use cases have managed to handle so far.

We can distinguish mainly two ways to establish trust on the Semantic Web: through the source and usage of the information, and through the information itself. Indeed, a key trend currently is about establishing practices that make it possible to trace where information on the

semantic web comes from, how it has been processed, and by whom. The Provenance Ontology [Lebo et al., 2013] is a key component of this approach, where the idea is that decisions on whether or not to trust information can be made easier by providing complete, structured data about its provenance [Moreau and Groth, 2013]. Ongoing work is also looking at mechanisms to trace back specific results of applications of semantic web data, by annotating the constituent processes of the applications by means of semantic relations between data artefacts [Daga et al., 2014]. While the availability of structured, semantic data to trace information back to its source is certainly a step forward, we are still far away from the ability to automate the process of establishing trust on the basis of such data, meaning that the selection of data and the decision to use knowledge from specific sources is still left to the developer of the system.

Besides assessing the source of information, one idea is of course to try and assess the information itself. Contrary to the previous approach, many of the approaches in this area have focused on automatization, especially by producing metrics aiming to provide automatic scores, as indicators of quality [Gangemi et al., 2006, Pipino et al., 2002]. Most of these approaches do not claim any ambition to provide a universally applicable information assessment method however, since, unsurprisingly, a common conclusion is that data quality is a subjective dimension, which is hard to evaluate out of context [Strong et al., 1997]. Intelligent semantic web systems are therefore bound to have to directly encode the context in which to assess information from the semantic web, as well as specific quality assessment methods.

Other types of information are also being used to try and understand how much trust a system can assign to specific pieces of data and knowledge from the semantic web. Taking inspiration from the human web, data and knowledge repositories [d'Aquin and Noy, 2012b] have implemented user rating systems, where the many users of the repositories can indicate to which extent they believe the information to be trustable, and this information can be represented in an automatically exploitable way [Lewen and d'Aquin, 2010]. This is of course, again, relying on the assumption that human users of the repositories have manually assessed the sources, and that their assessment is aligned with the requirements of the system.

When such user ratings are not available, other approaches have explored the way in which trust information could be indirectly obtained from either the usage of the data and knowledge, or their relationships with other sources. Taking inspiration from web-related mechanisms such as PageRank [Page et al., 1999], several approaches have looked into assessing quality through the way data is being reused or linked to other data, with the assumption that such reuse and links represented an indirect indication of approval [Ding et al., 2005]. Similarly, other approaches have looked into the way quality indicators can be obtained by comparing different sources, to detect errors, discrepancies, or biases [d'Aquin, 2009a, Liu et al., 2015, Sabou et al., 2007, Tiddi et al., 2014b]. Somehow, this leads to another form of scale issue: While the challenge for other approaches is to find methods to minimize the (human) effort required for assessing knowledge from the Semantic Web in the context of a specific system, here the challenge becomes that, especially compared to the "human web," the Semantic Web is not currently sufficiently large

and sufficiently used to enable the exploitation of redundancy and usage information for trust assessment. In other words, it has not reached a large enough scale and, because the nature of the Semantic Web is different from the one of the "Human Web," it might never do.

To summarize:

1. As obvious from discussions in previous chapters, the trust intelligent semantic web systems can assign to the quality of semantic web sources cannot be guarantied at the source.

2. Assessing quality and trust *a posteriori* is however very challenging, despite additional information sometimes being available about the provenance, usage, or rating of the information.

3. These issues are common on the Web already, but alleviated through exploiting its scale and redundancy. Similar approaches can be envisioned for intelligent semantic web systems, and research is already ongoing toward such approaches, but they can only be fully effective when the scale of the Semantic Web is high enough to guarantee redundancy.

4.2.2 POLICIES AND RIGHTS

Another area that, in principle, is much more formalized than data quality, but is nevertheless becoming increasingly challenging with the increase in size, variety, and complexity of knowledge and data on the semantic web, is the one concerned with the rights and policies attached to such knowledge and data. Under rights and policies, we consider in particular the licence attached to datasets, ontologies, and knowledge bases, distributed over the Web. A lot of work has gone into formalizing such licence and associated policies following the principles of the semantic web, in such a way that at least parts of the understanding and decisions could be subject to automatic processing and reasoning. At the center of such initiatives is the W3C ODRL community group[2] dedicated to the creation of a vocabulary for the representation of digital rights: ODRL (the Open Digital Rights Language; Iannella, 2002). ODRL is based on the basic principle that policies, e.g., data licence, can be represented through a set of actions, which can be either permitted, prohibited, or required. As an example, the code below (in RDF Turtle syntax) is a simplified representation of the Open Government License (OGL[3]) used by many open data providers:

```
<http://mksmart.org/dc/policy/ogl>
     a odrl:Set;
     odrl:permission odrl:copy ;
     odrl:permission odrl:publish ;
     odrl:permission odrl:reproduce ;
     odrl:permission odrl:commercialize ;
     odrl:duty odrl:attribute.
```

[2] https://www.w3.org/community/odrl/
[3] https://www.nationalarchives.gov.uk/doc/open-government-licence/version/3/

Although a rather simple solution, ODRL represents a key first step to support the manipulation of a certain kind of meta-aspects of knowledge and data on the Web, rights and policies, which would otherwise require large amounts of manual effort. Several initiatives have started to emerge that exploit this model to provide valuable resources and methods for more effective and scalable approaches to dealing with this issue, including an RDF dataset of common licence represented in ODRL (the RDF License dataset[4]), methods to apply deontic logics on data licence representations [Rotolo et al., 2013] and tools to support developers in selecting the right licence for the data they publish [Daga et al., 2015b].

However, if we consider again the template of intelligent semantic web systems drawn in Chapter 2, the idea that such systems could create a knowledge view from automatically selected, combined, and processed sources on the Web leads to issues that are much more complex than what can be addressed through the machine-processable information of policies on the semantic web. Indeed, when creating such a view from multiple sources there is a need to understand and reason upon the way policies of different sources affect each other, how they might or might not be altered through the selection, refactoring, integration process, and finally what (possibly inconsistent) requirements the intelligent semantic web system might have to satisfy to enforce these policies. Addressing these issues requires not only a representation of the policies that can be reasoned upon, but also a representation of the workflows that are applied to the integrated information (beyond what can be provided by PROV-O; see Daga et al., 2014), and of the rules that govern the propagation of policies across these workflows [Daga et al., 2015a]. This is especially challenging when, as on the semantic web, the boundaries of the artefacts (datasets, ontologies, etc.), to which these policies and licence apply, can be very fuzzy.

To summarize:

1. As technologies advance to enable intelligent semantic web systems to combine and process knowledge from large numbers of sources quickly and in large scales, dealing with the meta-information related to the rights to such knowledge might become a blockage.

2. Dealing with such meta-information means not only ways to represent and manage it, but also to reason upon it so that the propagation and validity of information policies can be automatically assessed in intelligent semantic web systems.

3. Part of such systems should therefore increasingly include mechanisms dedicated to the manipulation of right and policies, where the tasks of deciding and acting upon such information can be seen as an intelligent semantic web system in itself [d'Aquin et al., 2015].

4.2.3 PRIVACY

With a strong connection to data rights and policies, but representing potentially an even more challenging obstacle toward the widespread adoption of intelligent semantic web systems, are

[4] http://datahub.io/dataset/rdflicense

the issues of privacy and data protection. Indeed, in many countries, any application or service collecting or processing personal information is required to fulfill a number of requirements and follow a number of data protection and privacy principles.[5] Similarly to the issues associated with data licence and policies, there is very little understanding from developers of semantic web applications of these requirements and principles. As shown for example in d'Aquin et al. [2014b] on a small sample of such applications, most applications processing personal information, in one form or another, do not comply with even the most basic principles for the European Data Protection Act. This observation can be generalized to the systems described in Chapter 3, besides the ones developed by large companies with large organizational resources. In other words, while it can be argued that the Semantic Web and intelligent semantic web systems can reduce the technical effort required to build knowledge-intensive applications, they do not reduce the effort of dealing with privacy, making it even more complicated for developers without access to legal advice to reach the impact that their applications might promise. In addition, where ODRL has already provided a first step for the representation of machine processable policies, the many initiatives to formally represent privacy-related information (e.g., Sacco and Passant, 2011) have not yet resulted in the establishment of solid foundations for a semantic web-based approach to these issues. Pragmatically, the processes of ensuring compliance with the Data Protection Act, or of performing Privacy Impact Assessment [Wright and De Hert, 2011] are still far from including any form of automation.

Privacy in Intelligent Semantic Web Systems of course does not only affect the developers of applications, but also their users. Indeed, as already experienced with the growth of online services and social networks, connected systems that collect and integrate large quantities of information from many different sources, some of them private, lead to situations where the understanding and ability for individuals to manage the way such information is shared become impossible. This is especially true since, as argued for example in Solove [2008], privacy is itself already a complex notion to define, depending on the context and the purpose of providing a definition. As pointed out in Fried [1968]: "privacy is not simply an absence of information about us in the minds of others, rather it is the control we have over information about ourselves." To an extent, what is argued in the previous chapters is that intelligent semantic web systems are systems that are able to obtain, integrate, and reason upon information from the semantic web, without the need for manual intervention. Hence, there seems to be a strong inconsistency here, between intelligent semantic web systems acting on behalf of a user and this user managing her privacy by controlling the way information about herself is disclosed, shared, and processed. However, this is an inconsistency that semantic web technologies and intelligent semantic web systems might be better able to solve than any other approach. Indeed, the control of personal information is strongly related to the user's perception and understanding of their own privacy: what information users regard as private, from whom, and in what context, and depending on the privacy risks,

[5] See the UK Data Protection Act http://www.legislation.gov.uk/ukpga/1998/29/contents and the EU Data Protection Directive http://eur-lex.europa.eu/legal-content/en/NOT/?uri=CELEX:31995L0046.

how the users might compromise privacy for convenience, security, or commercial advantages. Thus, users' control of personal information is directly connected to the ability of the system to give them visibility, awareness, and accountability—the three properties of what is defined in Erickson and Kellogg [2000] as a "socially translucent system." Indeed, such systems are able "to support coherent behavior by making participants and their activities visible to one another." By nature, semantic web technologies possess key properties that might help achieving this: they focus on making information meaningful, and therefore transparent, through processing data and information about the system within the same model. A key challenge for intelligent semantic web systems is therefore to become mature enough to be able to support a coherent behavior with respect to privacy, the disclosure of and the control over personal information, by giving users a meaningful and transparent account of which information is processed in what way, to what purpose.

To summarize:

1. As with rights and policies, the ability of intelligent semantic web systems to create views that combine a large number of diverse sources of information makes the management of personal privacy immensely more complex, when these sources involve personal information.

2. Partly because of the focus on non-personal, open data, very little attention has been given by the semantic web research community to adapting already hard to deploy privacy-preserving approaches to intelligent semantic web systems.

3. If we consider privacy management as the ability to trace and make sense of our disclosures of personal information, and of their consequences, there is a lot of potential for research applying semantic web approaches to support such tasks.

4.2.4 INTERACTION

This particular challenge leads to another, more general issue for intelligent semantic web systems to tackle: *Interaction*. Indeed, there have been a lot of assumptions about the way individuals might interact with such systems. However, besides looking for example at specific issues to do with data and ontology visualization [Dadzie and Rowe, 2011], there does not seem to be currently a clear understanding about the way this new system paradigm should be adequately presented to users so that they can make the most of it. Early proposals in this area have made one of two assumptions. On the one hand, many researchers have assumed that the best way to interact with such systems was to reuse the interaction patterns established by earlier systems, adapting and extending them to the underlying principles of the semantic web. This approach covers the many Semantic Web and Linked Data browsers (e.g., Berners-Lee et al., 2006), which took the idea of surfing by means of a web browser as the most familiar interaction with the Web, and therefore created browsers to surf the web of data. While these systems have been useful in supporting people working on the

technical aspects of the semantic web, they have quickly shown their limitations when it came to interaction with end-users. Presenting information meant for machine consumption as if it were meant for human consumption is just not very effective.

Other approaches assumed from the start that, since the aim of the Semantic Web was to enable machines to intelligently process information, intelligent interfaces with human users would be necessary. In fact, the best example of this paradigm is the agent that helps a user to book hospital appointments, which was outlined in the original semantic web article [Berners-Lee et al., 2001]. This approach, where the intelligence of an intelligent semantic web system is not only in the way it processes data and knowledge from the semantic web, but also in the way it interacts with users is certainly very appealing, especially to the many semantic web practitioners with a background in Artificial Intelligence. If we look again at the examples in Chapter 3, we can see that some of the most successful systems there have used this approach. For example, IBM Watson is capable of understanding questions in natural language, and to act as an agent mediating the interaction between humans and its knowledge processing capabilities. On the contrary, others try to lower the barrier of entry (and of adoption) to the system by integrating the intelligent information processing they perform into the familiar interfaces of common tasks. Google's Knowledge Graph is a typical example where the Intelligent Semantic Web System presents itself as just another feature of a very common task, web search, without requiring the interaction with this task to change significantly. Somehow, the second approach seems more likely to become common. More than a challenge to be tackled, interaction with Intelligent Semantic Web Systems is an area where better practices and methodologies should slowly emerge and become more mature, as developers understand better how to place them within the context, habits, and needs of web users.

To summarize:

1. Early thinking around adapting typical interaction patterns common on the Web to interacting with intelligent semantic web systems have quickly shown to be ineffective.

2. Somehow more similar to Artificial Intelligent systems, intelligent semantic web systems either need to implement more complex interactions methods (reflecting the intelligence of the system), or to integrate seamlessly with existing, common tasks, as shown through the examples of Chapter 3.

3. Crucial research is therefore needed to develop methodologies for the creation of intelligent semantic web systems that ensure effective interaction with the users, especially to support their integration within familiar environments.

4. Indeed, it is expected that, as the technology becomes more and more common, more and more intelligent semantic web systems will materialise as features of more general systems, with the reliance on the semantic web remaining mostly implicit.

4.3 CONCLUSION

Although, after almost 15 years, the Semantic Web as a research area is still relatively young, it already has a complicated history, from its origins in Artificial Intelligence to its focus on knowledge engineering, and the turn a few years ago to data processing and management aspects (i.e., linked data). This history is reflected in the evolution of the development of intelligent semantic web systems. Looking at the Semantic Web Challenge,[6] for example, it appears that the community of semantic web developers has changed, or at least that the focus has changed. Early editions focused on the key challenges of heterogeneity, and the way to automatically inject meaning into applications, so to make them smarter and better able to intelligently support users. The winner of the Semantic Web Challenge in 2003 (CS Active Space; Shadbolt et al., 2004) was truly an intelligent semantic web system, which automatically integrates multiple sources of information from the Web in a local view so that this information can be processed and reasoned upon to enable effective exploration and sense-making of the academic computer science domain by the user. Following this however, a change of focus emerged, due to the growing realization that, to be a success, the Semantic Web had first to be relevant, to have an impact and to go beyond being just an "academic exercise." Much of the attention turned then to "leaner" systems, that could achieve simpler tasks in a more effective way by using distributed, easy to integrate data sources, focusing in particular on linked data sources. Despite this deliberate effort to generate impact having been an undeniable success, the community has not yet moved to the next step. To reuse a concept, the hype cycle, that has been much abused, intelligent semantic web systems have already well passed the peak of expectation and not yet reached the plateau of productivity. In other words, both the theoretical and the practical foundations have been addressed to the minimal extent that we can now start using the Semantic Web as a platform to build intelligent systems. These systems will increasingly integrate with more "standard" ones and contribute to the technical trends in other areas such as Big Data, Data Science, Analytics, and Knowledge Discovery. The key for semantic web researchers, practitioners, and developers is therefore to use this platform and these foundations to tackle the challenges described above, which, as intelligent semantic web systems become more ubiquitous, will impact well beyond the infrastructure of the semantic web.

[6] http://challenge.semanticweb.org/

Bibliography

Aamodt, A. and Nygard, M. (1995) "Different roles and mutual dependencies of data, information, and knowledge—an AI perspective on their integration." *Data & Knowledge Engineering*, 16(3), pp. 191–222. DOI: 10.1016/0169-023x(95)00017-m. 10

Antoniou, G. and van Harmelen, F. A. (2008) *Semantic Web Primer*, 2nd ed. The MIT Press. 5

Ayers, A. and Völkel, M. (2008) "Cool uris for the semantic web." Working Draft, W3C. 9

Baader, F., Ed. (2003) *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press. DOI: 10.1017/cbo9780511711787. 10

Baumeister, J., Puppe, F., and Seipel, D. (2004) "Refactoring methods for knowledge bases. Engineering Knowledge in the Age of the Semantic Web." Springer Berlin Heidelberg, pp. 157–171. 27

Bechhofer, S., Möller, R., and Crowther, P. (2003) "The DIG Description Logic Interface." Description Logics 81. 15

Berners-Lee, T., Hendler, J., and Lassila, O. (2001) "The semantic web." *Scientific American*, 284(5), pp. 28–37. DOI: 10.1038/scientificamerican0501-34. 1, 5, 48, 59

Berners-Lee, T. (2003) "WWW Past and Future. " `http://www.w3.org/2003/Talks/0922-rsoc-tbl/`. 3

Berners-Lee, T., Chen, Y., Chilton, L., Connolly, D., Dhanaraj, R., Hollenbach, J., Lerer, A., and Sheets, D. (2006) "Tabulator: exploring and analyzing linked data on the semantic web." In *Proc. of the 3rd International Semantic Web User Interaction Workshop*, vol. 2006. 22, 58

Bizer, C., Heath, T., and Berners-Lee, T. (2009) "Linked data—the story so far." *International Journal on Semantic Web and Information Systems*, 5(3), pp. 1–22. DOI: 10.4018/jswis.2009081901. 2, 5, 7

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008) "Freebase: a collaboratively created graph database for structuring human knowledge." In *Proc. of the 2008 ACM SIGMOD International Conference on Management of Data*, pp. 1247–1250. DOI: 10.1145/1376616.1376746. 30

Borgida, A. and Serafini, L. (2003) "Distributed description logics: Assimilating information from peer sources." In *Journal on Data Semantics I*. Springer Berlin Heidelberg, pp. 153–184. DOI: 10.1007/978-3-540-39733-5_7. 16

Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L., and Stuckenschmidt, H. (2003) "C-owl: contextualizing ontologies." In *The Semantic Web-ISWC 2003*. Springer Berlin Heidelberg, pp. 164–179. DOI: 10.1007/978-3-540-39718-2_11. 16

Buil-Aranda, C., Arenas, M., and Corcho, O. (2011) "Semantics and optimization of the SPARQL 1.1 federation extension." In *The Semanic Web: Research and Applications*. Springer Berlin Heidelberg, pp. 1–15. DOI: 10.1007/978-3-642-21064-8_1. 52

Carroll, J. J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A., and Wilkinson, K. (2004) "Jena: implementing the semantic web recommendations." In *Proc. of the 13th International World Wide Web Conference on Alternate Track Papers and Posters*. ACM, pp. 74–83. DOI: 10.1145/1013367.1013381. 22

Chu-Carroll, J., Fan, J., Boguraev, B. K., Carmel, D., Sheinwald, D., and Welty, C. (2012) "Finding needles in the haystack: Search and candidate generation." *IBM Journal of Research and Development*, 56(3.4), pp. 6–10. DOI: 10.1147/JRD.2012.2186682. 39

Conklin, J. (1987) "Hypertext: an introduction and survey." *Computer*, 20(9), pp. 17–41. DOI: 10.1109/mc.1987.1663693. 7

Cunningham, H. (2002) "GATE, a general architecture for text engineering." *Computers and the Humanities*, 36(2), pp. 223–254. DOI: 10.3115/993268.993365. 41

Dadzie, A. S. and Rowe, M. (2011) "Approaches to visualising linked data: a survey." *Semantic Web*, 2(2), pp. 89–124. DOI: 10.3233/SW-2011-0037. 22, 58

Daga, E., d'Aquin, M., Gangemi, A., and Motta, E. (2015a) "Propagation of Policies in Rich Data Flows," *K-Cap 2015*. DOI: 10.1145/2815833.2815839. 56

Daga, E. (2012) "Towards a theoretical foundation for the harmonization of linked data." *The Semantic Web—ISWC 2012*. Springer Berlin Heidelberg, pp. 445–448. DOI: 10.1007/978-3-642-35173-0_36. 19

Daga, E., d'Aquin, M., Gangemi, A., and Motta, E. (2014) "Describing semantic web applications through relations between data nodes." *KMi Technical Report*, pp. 14–15. `http://kmi.open.ac.uk/publications/techreport/kmi-14-05` 54, 56

Daga, E., d'Aquin, M., Motta, E., and Gangemi, A. (2015b) "A Bottom-up Approach for Licences Classification and Selection." *12th Extended Semantic Web Conference*. DOI: 10.1007/978-3-319-25639-9_41. 56

Daga, E., Panziera, L., and Pedrinaci, C. (2015c) "A BASILar approach for building Web APIs on top of SPARQL endpoints." In *Proc. of the SALAD Workshop at ESWC 2015*. 51

d'Aquin, M., Adamou, A., and Dietze, S. (2013a) "Assessing the educational linked data landscape." In *Proc. Web Science*. DOI: 10.1145/2464464.2464487. 23

d'Aquin, M., Lieber, J., and Napoli, A. (2013b) "Decentralized case-based reasoning and semantic web technologies applied to decision support in oncology." In *The Knowledge Engineering Review*, 28(04), pp. 425–449. DOI: 10.1017/s0269888913000027. 52

d'Aquin, M., Lieber, J., and Napoli, A. (2005) "Decentralized case-based reasoning for the semantic web." In *The Semantic Web–ISWC 2005*. Springer Berlin Heidelberg, pp. 142–155. DOI: 10.1007/11574620_13. 7

d'Aquin, M., Schlicht, A., Stuckenschmidt, H., and Sabou, M. (2007) "Ontology modularization for knowledge selection: Experiments and evaluations." In *Database and Expert Systems Applications*. Springer Berlin Heidelberg, pp. 874–883. DOI: 10.1007/978-3-540-74469-6_85. 25

d'Aquin, M. (2009a) "Formally measuring agreement and disagreement in ontologies." In *Proc. of the 5th International Conference on Knowledge Capture*. ACM, pp. 145–152. DOI: 10.1145/1597735.1597761. 25, 54

d'Aquin, M. and Lewen, H. (2009b) "Cupboard–a place to expose your ontologies to applications and the community." *The Semantic Web: Research and Applications*. Springer Berlin Heidelberg, pp. 913–918. DOI: 10.1007/978-3-642-02121-3_81. 23

d'Aquin, M., Euzenat, J., Le Duc, C., and Lewen, H. (2009) "Sharing and reusing aligned ontologies with cupboard." In *Proc. of the 5th International Conference on Knowledge Capture*. ACM, pp. 179–180. DOI: 10.1145/1597735.1597771. 23

d'Aquin, M. and Motta, E. (2011) "Watson, more than a semantic web search engine." *Semantic Web*, 2(1), pp. 55–63. 24, 49

d'Aquin, M., Ding, L., and Motta, E. (2011) "Semantic web search engines." In *Handbook of Semantic Web Technologies*. Springer Berlin Heidelberg, pp. 659–700. DOI: 10.1007/978-3-540-92913-0_16. 24, 30, 52

d'Aquin, M. (2012a) "Putting linked data to use in a large higher-education organisation." In *Proc. of the Interacting with Linked Data (ILD) Workshop at Extended Semantic Web Conference*. 35

d'Aquin, M., Allocca, C., and Collins, T. (2012) "DiscOU: a flexible discovery engine for open educational resources using semantic indexing and relationship summaries." In *11th International Semantic Web Conference*, p. 13. 35

d'Aquin, M. and Noy, N. F. (2012b) "Where to publish and find ontologies? A survey of ontology libraries." In *Web Semantics: Science, Services and Agents on the World Wide Web*, 11, pp. 96–111. DOI: 10.1016/j.websem.2011.08.005. 23, 54

d'Aquin, M., Davies, J., and Motta, E. (2015) "Smart cities' data: challenges and opportunities for semantic technologies." In *IEEE Internet Computing*, November/December 2015. DOI: 10.1109/mic.2015.130. 56

d'Aquin, M., Motta, E., Sabou, M., Angeletou, S., Gridinoc, L., Lopez, V., and Guidi, D. (2008a) "Toward a new generation of semantic web applications." *Intelligent Systems, IEEE*, 23(3), pp. 20–28. DOI: 10.1109/mis.2008.54. 1, 11, 14, 20, 23, 24, 50

d'Aquin, M., Adamou, A., Daga, E., Liu, S., Thomas, K., and Motta, E. (2014a) "Dealing with diversity in a smart-city datahub." In *Semantics for Smarter Cities Workshop at ISWC 2014*. 53

d'Aquin, M., Sabou, M., Motta, E., Angeletou, S., Gridinoc, L., Lopez, V., and Zablith, F. (2008b) "What can be done with the semantic web? an overview of watson-based applications." In *Proc. of the SWAP 2008 Workshop—Semantic Web Applications and Perspectives*. 29

d'Aquin, M., Thomas, K., and Graupe, S. (2014b) "Non-Technical Support and Guidance, LinkedUp Project Deliverable D3.3.1." `http://linkedup-project.eu/files/2014/11/LinkedUp_D3.3.1.pdf` 57

Ding, L., Pan, R., Finin, T., Joshi, A., Peng, Y., and Kolari, P. (2005) "Finding and ranking knowledge on the semantic web." In *The Semantic Web, ISWC 2005*. Springer Berlin Heidelberg, pp. 156–170. DOI: 10.1007/11574620_14. 54

Erickson, T. and Kellogg, W. A. (2000) "Social translucence: an approach to designing systems that support social processes." In *ACM Transactions on Computer-human Interaction (TOCHI)*, 7(1), pp. 59–83. DOI: 10.1145/344949.345004. 58

Euzenat, J. and Shvaiko, P. (2007) *Ontology Matching*. Springer. DOI: 10.1007/978-3-642-38721-0. 26

Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A., and Welty, C. (2010) "The AI behind Watson—the technical article." *The AI Magazine*. 40

Ferrucci, D. A. (2012) "Introduction to 'This is Watson'."*IBM Journal of Research and Development*, 56(3.4), p. 1. DOI: 10.1147/JRD.2012.2184356. 39

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and Berners-Lee, T. (1999) "Hypertext Transfer Protocol—HTTP/1.1." DOI: 10.17487/rfc7234. 7

Finin, T., Fritzson, R., McKay, D., and McEntire, R. (1994) "KQML as an agent communication language." In *Proc. of the 3rd International Conference on Information and knowledge Management*. ACM, pp. 456–463. DOI: 10.1145/191246.191322. 15

Finin, T., Ding, L., Pan, R., Joshi, A., Kolari, P., Java, A., and Peng, Y. (2005) "Swoogle: searching for knowledge on the semantic web." In *Proc. of the National Conference on Artificial Intelligence*, vol. 20, no. 4, p. 1682. Menlo Park, CA; Cambridge, MA, London, AAAI Press; MIT Press. 24

Fried, C. (1968) "Privacy [a moral analysis]." *Yale Law Journal*, 77:47593. DOI: 10.1017/cbo9780511625138.008. 57

Gangemi, A., Catenacci, C., Ciaramita, M., and Lehmann, J. (2006) *Modelling ontology evaluation and validation.* Springer Berlin Heidelberg, pp. 140–154. DOI: 10.1007/11762256_13. 26, 54

Gearon, P., Passant, A., and Polleres, A. (2013), "SPARQL 1.1 Update, W3C Recommendation." 22

Genesereth, M. R. and Nilsson, N. J. (1987) *Logical Foundations of Artificial Intelligence.* Morgan Kaufmann Publishers Inc., San Francisco, CA. 1

Gennari, J. H., Musen, M. A., Fergerson, R. W., Grosso, W. E., Crubézy, M., Eriksson, H., Noy, N. F., and Tu, S. W. (2003) "The evolution of Protégé: an environment for knowledge-based systems development." In *International Journal of Human-computer Studies*, 58(1), pp. 89–123. DOI: 10.1016/s1071-5819(02)00127-1. 21

Gómez-Pérez, A. (2004) *Ontology evaluation. Handbook on ontologies*. Springer Berlin Heidelberg, pp. 251–273. DOI: 10.1007/978-3-540-24750-0_13. 26

Haase, P., Lewen, H., Studer, R., Tran, D. T., Erdmann, M. d'Aquin, M., and Motta, E. (2008) "The neon ontology engineering toolkit. WWW." 21

Haase, P., Schmidt, M., and Schwarte, A. (2011) "The Information Workbench as a Self-service Platform for Linked Data Applications." In *ISWC.* 22

Harris, S., Ilube, T., and Tuffield, M. (2010) "Enterprise Linked Data as Core Business Infrastructure." In *Linking Enterprise Data*. Springer, pp. 209–219. DOI: 10.1007/978-1-4419-7665-9_10. 22, 43, 44, 45

Harth, A., Hogan, A., Delbru, R., Umbrich, J., O'Riain, S., and Decker, S. (2007) "Swse: Answers Before Links!" 24

Hartig, O., Bizer, C., and Freytag, J. C. (2009) *Executing SPARQL queries over the web of linked data*. Springer Berlin Heidelberg, pp. 293–309. DOI: 10.1007/978-3-642-04930-9_19. 51

Hartig, O. and Freytag, J. C. (2012) "Foundations of traversal based query execution over linked data." In *Proc. of the 23rd ACM Conference on Hypertext and Social Media*. DOI: 10.1145/2309996.2310005. 25

Heath, T. and Bizer, C. (2011) "Linked data: evolving the web into a global data space." In *Synthesis Lectures on the Semantic Web: Theory and Technology*, 1(1), pp. 1–136. DOI: 10.2200/s00334ed1v01y201102wbe001. 5, 9, 52

Hendler, J. (2001) "Agents and the semantic web." In *IEEE Intelligent Systems*, (2), pp. 30–37. DOI: 10.1109/5254.920597. 51

Hendler, J. (2007) "The dark side of the semantic web." In *Intelligent Systems, IEEE*, 22(1), pp. 2–4. DOI: 10.1109/mis.2007.17. 10, 18

Hitzler, P., Krotzsch, M., and Rudolph, S. (2011) "Foundations of Semantic Web Technologies." Chapman and Hall/CRC. 5

Horrocks, I., Patel-Schneider, P. F., and van Harmelen, F. (2003) "From SHIQ and RDF to OWL: The making of a web ontology language." In *Web Semantics: Science, Services and Agents on the World Wide Web*, 1(1), pp. 7–26. DOI: 10.1016/j.websem.2003.07.001. 1

Horrocks, I., et al. (2005) "Semantic web architecture: Stack or two towers?" In *Principles and Practice of Semantic Web Reasoning*. Springer Berlin Heidelberg, pp. 37–41. DOI: 10.1007/11552222_4. 2, 3

Iannella, R. (2002) "Open digital rights language (odrl) version 1.1," W3C Note. 55

Karger, D. and Schraefel, M. C. (2006) "The pathetic fallacy of RDF." In *Proc. of the 3rd Semantic Web User Interaction Workshop*, Athens, Georgia. 9

Kutz, O., Lutz, C., Wolter, F., and Zakharyaschev, M. (2004) "E-connections of abstract description systems." In *Artificial Intelligence*, 156(1), pp. 1–73. DOI: 10.1016/j.artint.2004.02.002. 16

Lakemeyer, G. and Levesque, H. J. (2000) *The Logic of Knowledge Bases*. MIT Press. 11, 14, 15

Laney, D. (2001), "3D Data Management: Controlling Data Volume, Velocity and Variety." *Gartner report*. 53

Lassila, O. and Swick, R. R. (1999) "Resource Description Framework (RDF) Model and Syntax Specification." 9

Lebo, T., Sahoo, S., and McGuinness, D. (2012) "Prov-o: the prov ontology. W3C working draft, May, 03 2012." In *World Wide Web Consortium*. `http://www.w3.org/TR/prov-o` 26

Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., and Zhao, J. (2013) "Prov-o: the prov ontology, W3C Recommendation." 54

Lei, Y., Uren, V., and Motta, E. (2007) "A framework for evaluating semantic metadata." In *Proc. of the 4th International Conference on Knowledge Capture*. ACM. DOI: 10.1145/1298406.1298431. 26

Lewen, H. and d'Aquin, M. (2010) "Extending open rating systems for ontology ranking and reuse." In *Knowledge Engineering and Management by the Masses*. Springer Berlin Heidelberg, pp. 441–450. DOI: 10.1007/978-3-642-16438-5_34. 26, 54

Liu, D., d'Aquin, M., and Motta, E. (2015) "Linked Data Fact Validation through Measuring Consensus, Workshop on Linked Data Quality." At *ESWC 2015*. 54

Lopez, V. and Motta, E. (2004) "Ontology-driven question answering in Aqualog." In *Natural Language Processing and Information Systems*. Springer Berlin Heidelberg, pp. 89–102. DOI: 10.1007/978-3-540-27779-8_8. 41

Lopez, V., Nikolov, A., Sabou, M., Uren, V., Motta, E., and d'Aquin, M. (2010) "Scaling up question-answering to linked data." In *Knowledge Engineering and Management by the Masses*. Springer Berlin Heidelberg, pp. 193–210. DOI: 10.1007/978-3-642-16438-5_14. 41

Lopez, V., Fernández, M., Motta, E., and Stieler, N. (2012) "PowerAqua: supporting users in querying and exploring the semantic web." In *Semantic Web*, 3(3), pp. 249–265. 25, 41

McCarthy, J. and Hayes, P. J. (1969) "Some philosophical problems from the standpoint of artificial intelligence. " *Readings in Artificial Intelligence*, pp. 431–450. DOI: 10.1016/b978-0-934613-03-3.50033-7. 13

McGuinness, D. L. and van Harmelen, F. (2004) "OWL web ontology language overview, W3C Recommendation." 10

Mendes, P. N., Jakob, M., García-Silva, A., and Bizer, C. (2011) "DBpedia spotlight: shedding light on the web of documents." In *Proc. of the 7th International Conference on Semantic Systems*. ACM, pp. 1–8. DOI: 10.1145/2063518.2063519. 35

Moreau, L. and Groth, P. (2013) *Provenance: An Introduction to PROV*. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool. DOI: 10.2200/s00528ed1v01y201308wbe007. 54

Morsey, M., Lehmann, J., Auer, S., and Ngonga Ngomo, A. C. (2011) "DBpedia SPARQL benchmark—performance assessment with real queries on real data." *The Semantic Web ISWC*, pp. 454–469. DOI: 10.1007/978-3-642-25073-6_29. 50

Newell, A. (1982) "The knowledge level." *Artificial Intelligence*, 18(1), pp. 87–127. DOI: 10.1016/0004-3702(82)90012-1. 5, 10, 14

Nikolov, A., Uren, V., Motta, E., and Roeck, A. (2008) "Integration of semantically annotated data by the KnoFuss architecture." In *16th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2008)*, Acitrezza, Italy. DOI: 10.1007/978-3-540-87696-0_24. 26

Nikolov, A., d'Aquin, M., and Motta, E. (2011) "What should I link to? Identifying relevant sources and classes for data linking." In *Joint International Semantic Technologies Conference*, Hangzhou, China. DOI: 10.1007/978-3-642-29923-0_19. 26

Nikolov, A., d'Aquin, M., and Motta, E. (2012) "Unsupervised learning of link discovery configuration." In *9th Extended Semantic Web Conference*, Heraklion, Greece. DOI: 10.1007/978-3-642-30284-8_15. 26

Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., et al. (2009) "BioPortal: ontologies and integrated data resources at the click of a mouse." In *Nucleic Acids Research*, 37(2), pp. W170–W173. DOI: 10.1093/nar/gkp440. 24

Oldakowski, R., Bizer, C., and Westphal, D. (2005) "RAP: RDF API for PHP." In *Proc. of the 1st Workshop on Scripting for the Semantic Web, 2nd European Semantic Web Conference*. 22

Oren, E. (2008) "Algorithms and components for application development on the semantic web." *Doctoral Dissertation*, National University of Ireland, Galway. 29

Osborne, F. and Motta, E. (2012) "Mining semantic relations between research areas." In *The Semantic Web—ISWC 2012*. Springer Berlin Heidelberg, pp. 410–426. DOI: 10.1007/978-3-642-35176-1_26. 46

Osborne, F., Motta, E., and Mulholland, P. (2013) "Exploring scholarly data with rexplore." In *The Semantic Web—ISWC 2013*. Springer Berlin Heidelberg, pp. 460–477. DOI: 10.1007/978-3-642-41335-3_29. 46

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999) "The PageRank citation ranking: bringing order to the Web." 54

Passant, A. (2010a) "Dbrec—music recommendations using DBpedia." In *The Semantic Web—ISWC 2010*. Springer Berlin Heidelberg, pp. 209–224. DOI: 10.1007/978-3-642-17749-1_14. 33, 34

Passant, A. and Decker, S. (2010b) "Hey! ho! let's go! explanatory music recommendations with dbrec." In *The Semantic Web: Research and Applications*. Springer Berlin Heidelberg, pp. 411–415. DOI: 10.1007/978-3-642-13489-0_34. 33

Passant, A. (2011), "Seevl: mining music connections to bring context, search and discovery to the music you like." In *Semantic Web Challenge*. 33

Pipino, L. L., Lee, Y. W., and Wang, R. Y. (2002) "Data quality assessment." In *Communications of the ACM*, 45(4), pp. 211–218. DOI: 10.1145/505999.506010. 54

Prud'Hommeaux, E. and Seaborne, A. (2008) "SPARQL query language for RDF, W3C Recommendation." 22

Prud'Hommeaux, E. and Seaborne, A. (2010) "Sparql 1.1 federation extensions, W3C Working Draft." 52

Pyle, D. (1999) *Data Preparation for Data Mining*. Vol. 1, Morgan Kaufmann. 26

R Development Core Team (2005) "A language and environment for statistical computing." In *R Foundation for Statistical Computing*. Vienna, Austria, 2013. `http://www.R-project.org`. 22

Raimond, Y., Abdallah, S. A., Sandler, M. B., and Giasson, F. (2007) "The music ontology." In *ISMIR*, pp. 417–422. 35

Rieß, C., Heino, N., Tramp, S., and Auer, S. (2010) "EvoPat—pattern-based evolution and refactoring of RDF knowledge bases." In *The Semantic Web—ISWC 2010*. Springer Berlin Heidelberg, pp. 647–662. DOI: 10.1007/978-3-642-17746-0_41. 27

Robinson, A. J. and Voronkov, A. Eds. (2001) *Handbook of Automated Reasoning*. Vol. 1. Elsevier. 22

Rohloff, K., Dean, M., Emmons, I., Ryder, D., and Sumner, J. (2007) "An evaluation of triple-store technologies for large data stores." In *On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops*. Springer Berlin Heidelberg, pp. 1105–1114. DOI: 10.1007/978-3-540-76890-6_38. 22

Rotolo, A., Villata, S., and Gandon, F. (2013) "A deontic logic semantics for licenses composition in the web of data." In *Proc. of the 14th International Conference on Artificial Intelligence and Law*. ACM, pp. 111–120. DOI: 10.1145/2514601.2514614. 56

Russell, S. J. and Norvig, P. (2003), "Artificial Intelligence: A Modern Approach," 2nd ed., Upper Saddle River, New Jersey, Prentice Hall. 1, 13

Sabou, M., Gracia, J., Angeletou, S., d'Aquin, M., and Motta, E. (2007) "Evaluating the semantic web: A task-based approach." Springer Berlin Heidelberg, pp. 423–437. 26, 54

Sabou, M., d'Aquin, M., and Motta, E. (2008) "Exploring the semantic web as background knowledge for ontology matching." In *Journal on Data Semantics XI*. Springer Berlin Heidelberg, pp. 156–190. DOI: 10.1007/978-3-540-92148-6_6. 26

Sacco, O. and Passant, A. (2011) "A privacy preference ontology (PPO) for Linked Data." In *LDOW*. 57

Scheuermann, A., Motta, E., Mulholland, P., Gangemi, A., and Presutti, V. (2013) "An empirical perspective on representing time." In *The 7th International Conference on Knowledge Capture*, ACM, Banff, Canada. DOI: 10.1145/2479832.2479854. 27

Schwarte, A., Haase, P., Hose, K., Schenkel, R., and Schmidt, M. (2011) Fedx: Optimization techniques for federated query processing on linked data. In *The Semantic Web ISWC 2011*. Springer Berlin Heidelberg, pp. 601–616. 52

Shadbolt, N., Motta, E., and Rouge, A. (1993) "Constructing knowledge-based systems." In *Software, IEEE*, 10(6), pp. 34–38. DOI: 10.1109/52.241964. 13

Shadbolt, N., Gibbins, N., Glaser, H., and Harris, S. (2004) "CS AKTive space, or how we learned to stop worrying and love the semantic web." In *IEEE Intelligent Systems*, (3), pp. 41–47. DOI: 10.1109/mis.2004.8. 20, 60

Shearer, R., Motik, B., and Horrocks, I. (2008) "HermiT: A Highly-Efficient OWL Reasoner." OWLED. Vol. 432. 22

Sirin, E., Parsia, B., Cuenca Grau, B., Kalyanpur, A., and Katz, Y. (2007) "Pellet: A practical owl-dl reasoner." In *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2), pp. 51–53. DOI: 10.1016/j.websem.2007.03.004. 1, 22

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., et al. (2007) "The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration." In *Nature Biotechnology*, 25(11), pp. 1251–1255. DOI: 10.1038/nbt1346. 24

Solove, D. J. (2008) *Understanding Privacy.* Harvard University Press. 57

Sowa, J. F. (1983) *Conceptual Structures: Information Processing in Mind and Machine.* Addison-Wesley Longman Publishing Co., Inc. Boston, MA. 14

Staab, S. and Stuckenschmidt, H. (2006) *Semantic Web and Peer-to-peer.* Springer-Verlag Berlin Heidelberg. DOI: 10.1007/3-540-28347-1. 51

Staab, S. and Studer, R., Eds. (2009) *Handbook on Ontologies.* Springer. DOI: 10.1007/978-3-540-92673-3. 1, 10

Strong, D. M., Lee, Y. W., and Wang, R. Y. (1997) "Data quality in context." In *Communications of the ACM*, 40(5), pp. 103–110. DOI: 10.1145/253769.253804. 54

Suárez-Figueroa, M. C., Gómez-Pérez, A., Motta, E., and Gangemi, A. Eds. (2012) *Ontology Engineering in a Networked World.* Springer. DOI: 10.1007/978-3-642-24794-1. 16, 17, 53

Tiddi, I., d'Aquin, M., and Motta, E. (2014a) "Dedalo: looking for clusters' explanations in a labyrinth of linked data." In *11th Extended Semantic Web Conference*, Crete. DOI: 10.1007/978-3-319-07443-6_23. 25, 52

Tiddi, I., d'Aquin, M., and Motta, E. (2014b) "Quantifying the bias in data links." In *Knowledge Engineering and Knowledge Management*. Springer International Publishing, pp. 531–546. DOI: 10.1007/978-3-319-13704-9_40. 52, 54

Tummarello, G., Delbru, R., and Oren, E. (2007) "Sindice.com: Weaving the open linked data." In *The Semantic Web*. Springer Berlin Heidelberg, pp. 552–565. DOI: 10.1007/978-3-540-76298-0_40. 24

van Harmelen, F., Ten Teije, A., and Wache, H. (2009) "Knowledge engineering rediscovered: towards reasoning patterns for the semantic web." In *Proc. of the 5th International Conference on Knowledge Capture*. ACM, pp. 81–88. DOI: 10.1145/1597735.1597750. 1, 14, 29

Vandenbussche, P. Y., Aranda, C. B., Hogan, A., and Umbrich, J. (2013) "Monitoring SPARQL endpoint status." In *International Semantic Web Conference (Posters and Demos)*, pp. 81–84. 51

Verborgh, R., Hartig, O., De Meester, B., Haesendonck, G., De Vocht, L., Vander Sande, M., and Van de Walle, R. (2014) "Querying datasets on the web with high availability." In *The Semantic Web ISWC 2014*. Springer International Publishing, pp. 180–196. DOI: 10.1007/978-3-319-11964-9_12. 51

Viljanen, K., Tuominen, J., and Hyvönen, E. (2009) "Ontology libraries for production use: The Finnish ontology library service ONKI." In *The Semantic Web: Research and Applications*. Springer Berlin Heidelberg, pp. 781–795. DOI: 10.1007/978-3-642-02121-3_57. 24

Volz, J., Bizer, C., Gaedke, M., and Kobilarov, G. (2009a) "Discovering and maintaining links on the web of data." In *The Semantic Web-ISWC 2009*. Springer Berlin Heidelberg, pp. 650–665. DOI: 10.1007/978-3-642-04930-9_41.

Volz, J., Bizer, C., Gaedke, M., and Kobilarov, G. (2009b) "Silk-A Link Discovery Framework for the Web of Data." In *LDOW*. 26

Vrandečić, D. (2009) *Ontology Evaluation*. Springer Berlin Heidelberg. DOI: 10.1007/978-3-540-92673-3_13. 26

Vrandečić, D. and Krötzsch, M. (2014) "Wikidata: a free collaborative knowledge base." In *Communications of the ACM*, 57(10), pp. 78–85. DOI: 10.1145/2629489. 30

Wright, D. and De Hert, P. (2011) *Privacy Impact Assessment*. Vol. 6. Springer Science and Business Media. DOI: 10.1007/978-94-007-2543-0. 57

Xu, L. and Embley, D. W. (2004) "Combining the Best of Global-as-View and Local-as-View for Data Integration." In *ISTA*, vol. 48, pp. 123–136. 26

Zimmermann, H. (1980) "OSI reference model—the ISO model of architecture for open systems interconnection." In *IEEE Transactions on Communications*, 28(4), pp. 425–432. DOI: 10.1109/tcom.1980.1094702. 4

Authors' Biographies

MATHIEU D'AQUIN

Mathieu d'Aquin is a Senior Research Fellow at the Knowledge Media Institute of The Open University. He obtained his Ph.D. in 2005 from the University of Nancy, France, where he worked on concrete applications of semantic technologies in the medical domain. He is now leading research around concrete solutions for the realization of intelligent systems producing and consuming knowledge from the semantic web, with applications in education, personal data management, and smart cities. Mathieu has authored more than 120 publications in leading journals and conferences in the field, and has led the development of some of the most innovative systems and services in the semantic web area, including the semantic web search engine Watson, `data.open.ac.uk`, the worldwide first linked data platform of a university, and the MK Data Hub, a smart city infrastructure for data curation, integration, and analysis. In 2011, he was recognized as one of the ten most promising young researchers in artificial intelligence, through the "AI 10 to watch" award from the prestigious magazine *IEEE Intelligent Systems*, and has won numerous other awards especially related to innovative applications of semantic technologies.

ENRICO MOTTA

Prof. Enrico Motta has a Ph.D. in Artificial Intelligence from The Open University in the UK, where he is currently a professor in Knowledge Technologies. In the course of his academic career he has authored over 300 refereed publications and his h-index is 58, an impact indicator that puts him among the top computer scientists in the world. His research focuses on large-scale data integration and analysis to support decision making in complex scenarios. He currently leads the MK:Smart project, a £16M initiative which aims to tackle key barriers to economic growth in Milton Keynes through the deployment of innovative data-intensive solutions in a number of sectors. He is also currently working on a novel environment for exploring and making sense of scholarly data, Rexplore, which leverages innovative techniques in large-scale data mining, semantic technologies, and visual analytics. Prof. Motta is also Editor-in-Chief of the *International Journal of Human-Computer Studies* and over the years has advised strategic research boards and governments in several countries, including the UK, the U.S., The Netherlands, Italy, Austria, Finland, and Estonia.