Social Networks and the Semantic Web

Péter Mika

December 18, 2006

SIKS Dissertation Series No. 2007-03.

The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Graduate School for Information and Knowledge Systems.

Promotiecommissie:
prof.dr. J. M. Akkermans (promotor, VUA/FEW)
prof.dr. T. Elfring (promotor, VUA/FSW)
dr. P. Groenewegen (co-promotor, VUA/FSW)
prof.dr. P. A. A. van den Besselaar (Universiteit van Amsterdam)
prof.dr. G. M. Duysters (Technische Universiteit Eindhoven)
prof.dr. J. A. Hendler (University of Maryland)
prof.dr. J. Kleinnijenhuis (Vrije Universiteit Amsterdam)
prof.dr. A. Th. Schreiber (Vrije Universiteit Amsterdam)
prof.dr. B. J. Wielinga (Universiteit van Amsterdam)

VRIJE UNIVERSITEIT

Social Networks and the Semantic Web

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad Doctor aan
de Vrije Universiteit Amsterdam,
op gezag van de rector magnificus
prof.dr. L.M. Bouter,
in het openbaar te verdedigen
ten overstaan van de promotiecommissie
van de faculteit der Sociale Wetenschappen
op maandag 5 februari 2007 om 10.45 uur
in de aula van de universiteit,
De Boelelaan 1105

door

Péter Mika

geboren te Boedapest, Hongarije

promotoren:	prof.dr. J.M. Akkermans
	prof.dr. T. Elfring
copromotor:	dr. P. Groenewegen

Preface

"A tie is anything you can tell a story about."
- Harrison White

It was a rainy winter day on February 13, 2004 when I arrived with a group of PhD students to the Dutch ski-resort town of Bergen (lit. *mountains*). We all signed up for what has been only referred to inside the university as the super-AiO course: two-days of training for PhD candidates (AiOs) in the art and science of graduating successfully and on time.

Our trainers, Brigitte and Jeanine have gone fairly confident about their work: after all, they had given the course before dozens of times. There was one difference this time, namely that all of us were foreigners and the course has been given in English for the first time. Nevertheless, our trainers spoke fluent English as most residents of Holland do, and thus expected few problems in dealing with our somewhat more diverse group.

Most of the first day has been spent with fun exercises such as directing each other blind-folded through the forest around the conference center. (An exercise in team building and communication, where failing and falling coincide.) This has lifted our spirits considerably and we didn't mind when something more serious came along: a lesson in planning, supposedly the skill most obviously in lack by most PhD students.

In particular, this was a part in the program where our trainers were trying to bring through the message that one can only reach his or her goals in life by drawing up some kind of a master plan. So we were asked to draw an image of where we would like to be in thirty years from now. (Example: famous professor with lots of free time.) We had to reason backwards from there, i.e. where do you need to be in ten years to get there etc., the idea being that if you follow this chain of reasoning, you can arrive to three concrete steps you can take right now to get closer to your goals in thirty years time. They would ask us to write down these steps on a postcard; they would mail these cards to us in three months time so that we would be confronted with the (lack of the) fulfilment of our promises.

The reaction from our group was a strong and immediate form of protest that took our trainers utterly by surprise. And then and there was the moment when I also realized what binds us together beyond our differences: the experience that life is not something you plan.

In fact, probably the only thing we had in common was that at one point or another we all took a step into the unknown. We have decided to pursue a PhD in the Netherlands,

leaving friends and family behind. A carefully considered step in the right direction: yes. With fully predictable consequences: hardly. We didn't deny the usefulness of some planning. And we are not necessarily adventurers either. But based on our life experience, all of us seemed to agree that a life that follows a master plan without unforeseen risks and sidetracks is not one we would like to live.

In June 2004, the word 'serendipity' has been voted as one of the top ten English words that are hardest to translate. Nevertheless, it comes closest to describe the experience of finding valuable things not by explicitly looking, but rather by generally walking along the right path, keeping an open mind and yes, having some luck sometimes.

Partly chosen, partly given, I feel that serendipity has served me well in the past five years. The road I've chosen took me through a one-year master's program, a half-year internship at Aidministrator (now Aduna) and four years of PhD at the ever unofficial Semantic Web Group of the VUA. Along the way I have met a wonderful group of people who supported me, guided me and contributed to every step I made. I have learned that science of any kind is first and foremost about people, and it may not be incidental that this is also a subject of this thesis...

First is my second family of friends and colleagues. I owe them that I can now honestly say: I would not have liked to spend my past years in any other way. I'll not list your names at the risk of forgetting some... I will miss you. My only consolation is that (as my own website reminds me) our networks in science are well able to extend across geographical and organizational boundaries (and if Skype doesn't work, you can always come to visit me).

Speaking of my thesis, it certainly would not have been made possible without my first promotor, Hans Akkermans. He put a great deal of trust in me, while looking out for me all along. I'm grateful in particular for giving me the freedom to explore the Semantic Web realm before settling on the topic of this thesis. As expected from a supervisor, he shared his significant experience in conducting research, being a researcher and dealing with researchers. However, it is through his personality, above and beyond all, that he conveyed the most important meta-lessons of research: that you should not be afraid to fail and that progress is made by looking beyond the conventional, the ordinary and the boring.

Tom Elfring and Peter Groenewegen joined Hans in my supervision once I started on my interdisciplinary research within the newly established VU Research School for Business Information Sciences (VUBIS). They have been my guides through the wonderland of Social Science, a world remarkably different from my home base in Computer Science. It's been a true exchange with Tom and Peter. I'm most grateful for what I've learned, for all their effort in trying to understand the technological details of my work and for working with me to translate the outcomes into concrete and relevant results in the field of applied network analysis.

In terms of the number of discussions, a special mention goes to Frank van Harmelen. I've counted them and Frank is the person I've exchanged the most emails with in the past years. And then I didn't count the number of discussions on the corridor, during lunch or traveling to or from somewhere. Simply, he is the kind of role model that every PhD student should have in the vicinity.

Lastly, there are important people in my personal life I would like to acknowledge here. First, my mother who passed away too early to see me graduate. Then the rest of my family who had to miss me for long periods and then had to be satisfied with seeing me for only a couple of days at a time: *köszönöm, hogy mellettem álltatok és segítettetek!* I will never be able to thank enough Andor, who undoubtedly made the most personal sacrifices for my PhD. Last, but not least I owe Dirk the world for his unwavering love in the past years. And hopefully for more to come...

Péter Mika
Amsterdam
December 15, 2006

Contents

Chapter 1

Introduction

Modern day research is faced with both extraordinary opportunities and challenges. On the one hand, a fast paced modern society turns to academics as public servants for immediate answers to the practical problems created by its own increasing needs and desires. Society is willing to invest in research as the basis of a knowledge economy as long as research proves to be responsive to its needs.

On the other hand, most of the questions science is required to answer are too complex to be addressed in the traditional disciplinary framework of academic research. Yet with the explosion of knowledge, research has become only more fragmented than ever. The lack of communication even within the faculties of a single university leads to opportunities lost every single day. (And even research groups within universities become specialized: long gone are the days where every subdiscipline within a scientific domain was equally represented at a university.)

Most apparent is the way the extraordinary technological developments of the computer age are left confined to the area of Computer Science. A few successes such as the combination of Biology and Informatics in the field of Bioinformatics show us a glimpse of the unprecedented capabilities of a multidisciplinary approach in addressing problems that have been thought to be too difficult only a decade ago due to the human effort involved. Replacing human effort with computing power led to discoveries such as the map of the human genome. And such interaction is not just a one-way technology transfer from Computer Science to an application area: information science itself is shaped increasingly by ideas borrowed from application areas such as the biological world.

The research contained in this thesis answers some specific research questions in Computer Science and Social Science and proves that an interdisciplinary approach brings appropriate results and contributes to our understanding of both fields. On the one hand we address theoretical questions and practical problems in Social Network Analysis by using the methods of Computer Science in the process of data collection, management and presentation. On the other hand, we apply the world view and methods of Network Analysis in understanding the role of social networks in the technology that underlies the Semantic Web, a technological innovation that will lead us to a next generation of the World Wide Web.

This work would not have been possible without a new form of academic research funding at the Vrije Universiteit Amsterdam (VUA), which instigates interdisciplinary research and puts the necessary structures in places. In 2003, the VU has begun implementing its new long term research vision by providing special funding for interdisciplinary research. The Vrije Universiteit Research School for Business Information Sciences (VUBIS) has come into existence as a collaboration between the Faculties of Science (FEW), Social Science (FSW), Economics and Business Administration (FEWEB). The VUBIS initiative obtained initial funding for eight interdisciplinary PhD projects, including the one discussed in this thesis.

1.1 Social Networks and the Semantic Web

The *Semantic Web* is a term coined by Tim Berners-Lee for the next stage in the evolution of the World Wide Web he initiated. In this vision of the future Web, information is given well-defined meaning (semantics) in a way that allows our computers to combine and reason with information from multiple sources just as we do ourselves when we search and browse the Web. Since our machines have only limited ways to access the semantics of information at the moment (to understand the content of text, images etc.), additional formal descriptions need to be provided for the information and services populating the web.

For the sake of interoperability such descriptions have to be expressed in shared, formal and domain-specific vocabularies: these *ontologies* capture the agreement within a community over the set of concepts and relationships in the domain and contain logic-based descriptions of these. In order to prevent the segmentation of the Semantic Web into islands of semantics, users and communities are also given the possibility to use and extend each other's ontologies, forming a Web of ontologies and metadata. Although there is still a scarcity of semantics on the Web, the idea of the Semantic Web has inspired a new stream of research of applying the results of Knowledge Representation in the setting of the Web as well as in other scenarios (e.g. enterprise systems).

In our primary study, we apply Semantic Web technology to the aggregation of the electronic data sets that we collect about the social networks of researchers working on the realization of the Semantic Web. We analyze these data using the methods of network analysis and make contributions to the field of scientometrics by measuring the impact of social networks on the success (or failure) of researchers.

In our secondary study, we look at the role of social networks within the architecture of the Semantic Web. Although this has been largely overlooked by the early proponents of the Semantic Web, it is now apparent that the Semantic Web is as much a socio-technological innovation as a purely technological one. While the expectation was that the Semantic Web would function with highly formal ontologies with minimal ambiguity and thus a minimal need for human interpretation, we now encounter the limitations of increasing the formality of knowledge in the global, dynamic environment of the Web.

As a result, much of the semantics in the *lightweight ontologies* we find is not part of formal agreements but implicit in the way ontological terms are used by a certain community of users. In particular, the *folksonomies* or tagging systems that are the basis

of many novel knowledge sharing applications developed under the banner of *Web 2.0* are too poor to be understood in the logical framework of the Semantic Web. The methods of network analysis on the other hand can be effectively applied to recover the shared, implicit meaning behind the terms in folksonomies by looking at their usage patterns within certain communities.

1.2 Research Questions

We propose the following two independent research questions, each with a number of sub-questions.

- How does the relational and cognitive structure of social networks affect the substantial outcomes of emerging scientific-technological communities such as the Semantic Web?
  - Can we exploit the Web as a data source for social network analysis?
  - How could we assess the reliability of network data obtained from the Web?
  - How could we support the reuse and aggregation of electronic data in complex studies in Social Network Analysis?
- What is the computational role of social networks in the emergence of semantics in folksonomies?
  - Can we conceptualize folksonomies as lightweight, dynamic, socially grounded ontologies?
  - If yes, how can we extract the semantics of terms emerging through usage?

The primary research question is relevant to the Social Sciences, in particular to understanding the effects of social networks on innovation and science. Our sub-questions are motivated by the technological opportunities obtaining large scale data for network analysis from the Web and applying Semantic Web technologies in the management of social network data.

The secondary research question is relevant to Information and Computer Science, in particular to the development of new forms of emergent ontologies for Semantic Web. It is motivated by the close tie between social networks and cognitive similarity and builds on available data in the form of large scale lightweight social-semantic networks (folksonomies).

1.3 Research Methodology

Answering interdisciplinary queries such as the ones posed above requires an extended world view and approach. In terms of world view, the researcher needs to transcend the boundaries of single disciplines and acquire knowledge of the different domains of investigation (information systems on the one hand and social systems on the other) and

their connectivity. Based on this new world view existing methods of investigation may also need to be combined into a new, interdisciplinary methodology.

In disciplinary works of Social Science, the objects of study are real world objects; in the case of network analysis the focus is on the social ties and social groups that make up the structure of human communities. Two phases of research constitute the accepted methodology of the research field. In the first phase of research development the researcher is required to generate hypotheses about social structures by observing them in the real world. The dominating methods for this kind of study are qualitative, including direct observation (field study) and unstructured interviews. In the second phase of research development, the researcher is required to verify the hypothesis by testing it using quantitative methods, in the same or different context. It is common that the two phases of research are executed by different persons, in particular the same hypothesis is typically tested in different settings before it becomes an accepted part of scientific knowledge.

In comparison, the world of Information and Computer Science is populated by information artifacts; the fact that some of these information artifacts may represent real world objects (e.g. an eight-by-eight matrix of bytes used to represent a chess table) is of limited concern to the information sciences: once an informational representation of a real world phenomenon is established (i.e. we agree that an eight-by-eight matrix of bytes captures all important aspects of a chess puzzle), the link between the model and the real world can be ignored. The task of the researcher is typically the engineering of abstract methods (algorithms) that transform information artifacts in desirable ways, e.g. apply the rules of chess to come up with a solution to the puzzle. The properties of the algorithms are also of interest, e.g. whether a solution is found for all puzzles with a solution (completeness) or what computational properties they have (e.g. time and space complexity.)

In this thesis we take an interdisciplinary, holistic view of the world when treating both of our research questions. In this view the informational world is not a separate distillate of the real world but constituted by it. This also means that the link between the two cannot be ignored. In fact, the link between the two worlds serves as a major inspiration for our work.

As we have observed in our work, acknowledging this world view takes a leap of faith from the disciplinary researcher. This is most visible where domains and methods cross disciplinary domains and these points can be easily identified.

In terms of Social Science, the leap of faith concerns the use of online data in place of real world data. While we return to this point later, we note that even today researchers in this area feel compelled to cite the early work of Wellman [Wellman et al., 1996], which for them sanctions the use of *electronic data* such as data from the Web. Discussions about the particulars such as how well the Web reflects publication networks are still debated, however. The caution towards online data is least prevalent in the methodological core of network analysis community[1], but more strongly felt in the application areas of network analysis such as Organization Science and Management Science. As we might

[1] Sessions about the analysis of online networks have been part of the International Sunbelt Social Network Analysis conference series at least since 1998. The Sunbelt community itself has also embraced the use of the internet early on with a regular mailing list, educational website and online journal.

expect, this is no concern to the computer scientists (like the author of this thesis), who are glad to mine the Web for real world knowledge.

In terms of Computer Science, the leap of faith is required to understand the role of social networks as part of the Semantic Web architecture. In particular, computer scientists are predisposed to a view of knowledge as an abstract artifact that can be detached from the social context. While again we return to this argumentation later, we note here that the developments in the last year of this PhD point to a growing acceptance of the role of social networks.[2]

1.4 Relevance to Social Science

1. Theoretical contributions

   (a) This thesis provides a **methodology** based on Semantic Web technology for supporting the full process of extracting, representing, aggregating, storing and visualizing social network data. Semantic Web technology plays a particular role in supporting the **aggregation of data** from heterogeneous sources of information, leading to more robust and more easily comparable results in network analysis.

   (b) Further, we explore the possibility of applying Web mining to social network **data acquisition** from the Web in order to support large scale, longitudinal studies in network analysis. We improve on existing methods and for the first time provide an **evaluation of web-based extraction** in comparison to the survey method of data collection most commonly used in Social Network Analysis.

   (c) Lastly, we provide theoretical contributions by applying our methodology to the field of **scientometrics**. Based on a large scale, multi-source data set we test the positive impacts of a structurally and cognitively diverse personal networks as measured by the performance of researchers within the Semantic Web community.

2. Practical results

   (a) Both the **implementation of our methods** and the **data set** collected in this thesis are available as open source for conducting further experiments in the same or different domains. We demonstrate our results through the **Flink website**, which displays the social networks and research profiles of members of the Semantic Web research community. As an application of Semantic Web technology, Flink has been awarded a first prize at the Semantic Web Challenge of 2004.[3]

[2] In particular, the International Semantic Web Conference (ISWC) in 2005 has seen a number of workshops dedicated to the topic as well as a session devoted to social networks and the Semantic Web. At the same time, the EU has also supported the startup of a significant project on this topic with close to 50 million euros of funding.

[3] See `http://challenge.semanticweb.org`

1.5 Relevance to Information and Computer Science

1. Theoretical contributions

   (a) Driven by the requirements of data aggregation in the science domain, we explore the general problem of data heterogeneity at the instance level. While the more common methods of ontology mapping target the mapping of ontologies on the schema level, the goal of **instance unification** or *smushing* is to map instances that denote the same real world entity. We explore the requirements of smushing from the representation side and discuss ways to use existing tools (query engines and reasoners) for performing the mapping.

   (b) Based on the observation that social networks influence how we conceptualize the world, we further explore **the social dimension of semantics** in a study of *folksonomies*, lightweight semantic structures where the semantics of terms is largely implicit in their usage. We propose a representation of folksonomies that incorporates the actor who is making annotations and show how methods of network analysis can be used to uncover the semantics emerging through usage. We also show that the perceived correctness of emergent models is in fact dependent on the social context. This work has been awarded a Best Paper award at the International Semantic Web Conference (ISWC) of 2005.

2. Practical results

   (a) The **implementation of our methods** for instance unification is available as part of the open source Elmo package, an API for the popular Sesame ontology storage facility.[4]

1.6 Structure of this Thesis

Every thread in this thesis leads to one or more of the three major studies described in Chapters 8, 9 and 10. In Chapter 8, we compare the results of novel methods of social network extraction from the Web with the outcomes of a survey in a research community. We use data from social network mining and other sources in our scientometric study of the Semantic Web research community in Chapter 9. We explore the role of the social context in knowledge representation for the Semantic Web in Chapter 10.

Chapters 4, 5, 6 and 7 contain the necessary details that would allow anyone to reproduce our work and to design and execute social studies where data are collected automatically and aggregated using semantic technology.

Chapters 2 and 3 introduce the key concepts of the Semantic Web and Social Network Analysis. These chapters prepare both computer scientists with an interest in networks and social scientists with an interest in electronic data for understanding the subsequent discussions.

[4] See `http://www.openrdf.org`

1.7 Publications

This thesis is based on and has led to the following list of international refereed publications:

- Hans Akkermans and Peter Mika. Ontology Technology, Knowledge Articulation, and Web Innovation, chapter in: *Advances in Knowledge Management Vol. III*, Jos Schreinemakers et al. (eds.), Ergon Verlag, 2006
- Peter Mika. Ontologies are us: A unified model of social networks and semantics. *Journal of Web Semantics* 4 (4), 2006. To Appear.
- Peter Mika, Tom Elfring and Peter Groenewegen. Application of semantic technology for social network analysis in the sciences. *Scientometrics* 68 (1), page 3–27, 2006
- Peter Mika. Ontologies are us: A unified model of social networks and semantics. In: *Proceedings of the Fourth International Semantic Web Conference (ISWC 2005)*, Yolanda Gil, Enrico Motta, Richard V. Benjamins and Mark Musen (eds.) , Lecture Notes in Computer Science no. 3729, page 122–136, Galway, Ireland, November, 2005. Winner of the Best Paper Award at ISWC 2005.
- Peter Haase, Bjorn Schnizler, Jeen Broekstra, Marc Ehrig, Frank van Harmelen, Maarten Menken, Peter Mika, Michal Plechawski, Pawel Pyszlak, Ronny Siebes, Steffen Staab and Christoph Tempich. Bibster – A Semantics-Based Bibliographic Peer-to-Peer System. *Journal of Web Semantics* 2 (1), page 99–103, 2005
- Peter Mika. Flink: Semantic Web Technology for the Extraction and Analysis of Social Networks. *Journal of Web Semantics* 3 (2), page 211–223, 2005. Winner of the Semantic Web Challenge of 2004.
- Peter Mika. Social Networks and the Semantic Web: The Next Challenge. *IEEE Intelligent Systems* 20 (1), January/February, pages 80–93, 2005
- Peter Haase, Jeen Broekstra, Marc Ehrig, Maarten Menken, Peter Mika, Michal Plechawski, Pawel Pyszlak, Bjorn Schnizler, Ronny Siebes, Steffen Staab and Christoph Tempich. Bibster – A Semantics-Based Bibliographic Peer-to-Peer System. In: *Proceedings of the Third International Semantic Web Conference (ISWC 2004)*, Sheila A. McIlraith, Dimitris Plexousakis and Frank van Harmelen (eds.), page 122–136, Hiroshima, Japan, November, 2004
- Peter Mika. Social Networks and the Semantic Web: An Experiment in Online Social Network Analysis. In: *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, pages 285–291, Beijing, China, September, 2004
- Peter Mika. Bootstrapping the FOAF-web: An Experiment in Social Network Mining. In: *1st Workshop on Friend of a Friend, Social Networking and the Semantic Web*, Galway, Ireland, September, 2004

- Peter Mika and Aldo Gangemi. Descriptions of Social Relations. In: *Proceedings of the 1st Workshop on Friend of a Friend, Social Networking and the (Semantic) Web*, 2004

- Peter Mika and Hans Akkermans. Towards a New Synthesis of Ontology Technology and Knowledge Management. *Knowledge Engineering Review* 19 (4), page 317–345, 2004

- Peter Mika, Marta Sabou, Aldo Gangemi and Daniel Oberle. Foundations for DAML-S: Aligning DAML-S to DOLCE. In: *Proceedings of First International Semantic Web Services Symposium (SWS2004)*, AAAI Spring Symposium Series, 2004

- Peter Mika, Daniel Oberle, Aldo Gangemi and Marta Sabou. Foundations for Service Ontologies: Aligning OWL-S to DOLCE. In: *Proceedings of the 13th International World Wide Web Conference (WWW2004)*, pages 563–572, 2004

- Aldo Gangemi and Peter Mika. Understanding the Semantic Web through Descriptions and Situations. In: *On The Move 2003 Conferences (OTM2003)*, Robert Meersman, Zahir Tari and Douglas Schmidt et al. (eds.), pages 689–706, 2003

- Peter Mika, Victor Iosif, York Sure and Hans Akkermans. Ontology-based Content Management in a Virtual Organization, chapter in: *Handbook on Ontologies in Information Systems*, Steffen Staab and Rudi Struder (eds.) , International Handbooks on Information Systems, page 447–471, 2003

- Christiaan Fluit, Herko ter Horst, Jos van der Meer, Marta Sabou and Peter Mika. Spectacle, chapter in: *Towards the Semantic Web: Ontology-Driven Knowledge Management*, ISBN 0-470-84867-7, 2003

- Victor Iosif, Peter Mika, Rikard Larsson and Hans Akkermans. Field Experimenting With Semantic Web Tools In A Virtual Organization, chapter in: *Towards the Semantic Web: Ontology-Driven Knowledge Management*, ISBN 0-470-84867-7, 2003

- Peter Mika. Integrating Ontology Storage and Ontology-based Applications Through Client-side Query and Transformations. In: *Proceedings of Evaluation of Ontology-based Tools (EON2002)* workshop at EKAW2002, Siguenza, Spain, 2002

- Peter Mika. JavaServer Pages (In Hungarian.), chapter in: *The J2EE Guide for Java Programmers*, J. N. Gaizler (eds.), page 115–163, ISBN 963-463-578-4, 2002

Part I

The Semantic Web and Social Networks

Chapter 2

The Semantic Web

The Semantic Web is the application of advanced knowledge technologies to the Web and distributed systems in general.

But why would the Web need any extension or fixing? We will argue that the reason we do not often raise this question is that we got used to the limitations of accessing the vast information on the Web. We learned not to expect complete or correct answers and not to ask certain questions at all.

In the following, we will demonstrate this effect on the example of some specific queries (Section 2.1). What is common to these questions is that in all cases there is a knowledge gap between the user and the computer: we are asking questions that require a deeper understanding of the content of the Web on the part of our computers or assume the existence of some background knowledge. As our machines are lacking both our knowledge and our skills in interpreting content of all kinds (text, images, video), the computer falls short of our expectations when it comes to answering our queries.

Knowledge technologies from the field of Artificial Intelligence provide the necessary means to fill the knowledge gap. Information that is missing or hard to access for our machines can be made accessible using *ontologies*. As we will see in Section 2.3, ontologies are in part social, part technological solutions. On the one hand, ontologies are formal, which allows a computer to emulate human ways of reasoning with knowledge. On the other hand, ontologies carry a social commitment toward using a set of concepts and relationships in an agreed way.

As Tim Berners-Lee describes in his book on the origin of the Web, the success of the Web hinged on social adoption as much as technological issues [Berners-Lee et al., 1999]. The Semantic Web adds another layer on the Web architecture that requires agreements to ensure interoperability and thus social adoption of this new technology is also critical for an impact on the global scale of the Web. As the Semantic Web community is also the subject of this thesis we will describe the development of the Semantic Web from its recent beginnings in Section 2.4. We discuss the recent parallel and complementary development of Web technologies known as Web 2.0 in Section 2.5.

We will enter into the details of ontology-based representation, the core of Semantic Web technology in Chapter 5. In Chapter 6 we will show how to use Semantic Web

technology for the management of data sources in the social domain, which we later apply in our case study of the Semantic Web community in Chapter 9.

2.1 Questions and answers

There is a general consent that the Web is one of the greatest inventions of the 20th century. But could it be better?

The reason that we do not often raise this question any more has to do with our unusual ability to adapt to the limitations of our information systems. In the case of the Web this means adaptation to our primary interface to the vast information that constitutes the Web: the search engine. In the following we list four questions that search engines cannot answer at the moment with satisfaction or not at all.

2.1.1 What's wrong with the Web?

The questions below are specific for the sake of example, but they represent very general categories of search tasks. As we will see later they also have in common that in each of these cases semantic technology would drastically improve the computer's ability to give more appropriate answers (Section 2.2).

1. **Who is Frank van Harmelen?**

   To answer such a question using the Web one would go to the search engine and enter the most logical keyword: *harmelen*. The results returned by Google are shown in Figure 2.1. (Note that the results are slightly different depending on whether one enters Google through the main site or a localized version.)

   If this question and answer would be parts of a conversation, the dialogue would sound like this:

   > Q: *Who is Frank van Harmelen?*
   >
   > A: *I don't know but there are over a million documents with the word "harmelen" on them and I found them all really fast (0.31s). Further, you can buy Harmelen at Amazon. Free Delivery on Orders Over 15.*

   Not only the advertizement makes little sense, but from the top ten results only six are related to the Frank van Harmelen we are interested in. Upon closer inspection the problem becomes clear: the word Harmelen means a number of things. It's the name of a number of people, including the (unrelated) Frank van Harmelen and Mark van Harmelen. Six of the hits from the top ten are related to the first person, one to the latter. Harmelen is also a small town in the Netherlands (one hit) and the place for a tragic train accident (one hit).

   The problem is thus that the keyword *harmelen* (but even the term *Frank van Harmelen*) is polysemous. The reason of the variety of the returned results is that designers of search engines know that users are not likely to look at more than the top ten results. Search engines are thus programmed in such a way that the

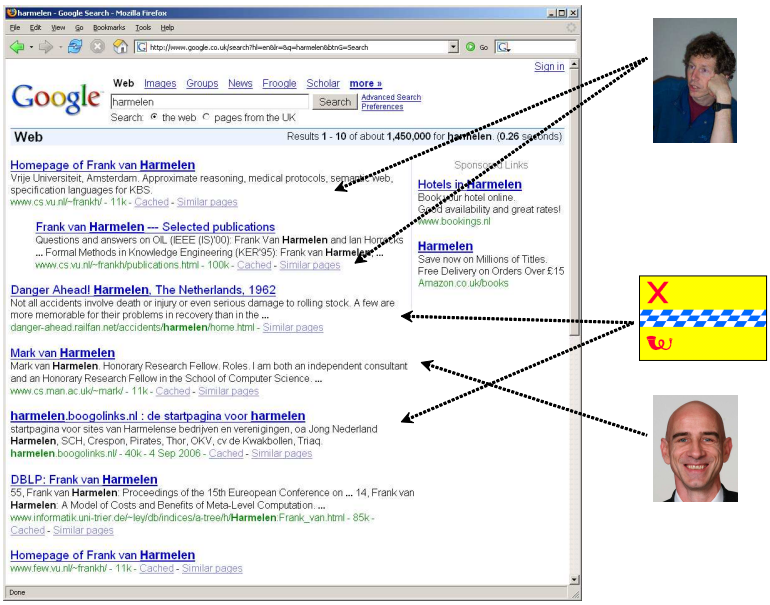


Figure 2.1: Search results for the keyword *harmelen* using Google.

first page shows a diversity of the most relevant links related to the keyword. This allows the user to quickly realize the ambiguity of the query and to make it more specific.

Studying the results and improving the query, however, is up to the user. This is a task we take for granted; in fact, most of us who are using search engines on a daily basis would expect this confusion to happen and would immediately start with a more specific query such as *Frank van Harmelen*. While this excludes pages related to the municipality of Harmelen, it is important to note that this would not solve our problem completely. If we browse further in the results we notice that the overwhelming majority of the results are related to prof. Frank van Harmelen of the Vrije Universiteit, but not all of them: there are other people named Frank van Harmelen. (In fact, finding them would be a lot more difficult: all of the high ranking pages are related to prof. Harmelen, who has a much larger representation on the Web due to his work related to Semantic Web technology.)

Again, what we experience is an ambiguity of our query that we could solve by adding additional terms such as *Vrije Universiteit* or *research*. This leads to another problem: our request becomes overspecified. First, it is not guaranteed that every mentioning of Frank van Harmelen is accompanied by any or all of these words. Worse yet, pages about Frank van Harmelen may not even mention him by name. None of our queries would return pages about him where he is only mentioned by his first name for example or as *van Harmelen, F.* Not even if for the

human reader it would be blatantly obvious that the Frank in question could only be Frank van Harmelen.

2. **Show me photos of Paris**

   The most straightforward solution to this search task is typing in "paris photos" in the search bar of our favorite search engine. Most advanced search engines, however, have specific facilities for image search where we can drop the term photo from the query. Some of the results returned by Google Image Search are shown in Figure 2.2.

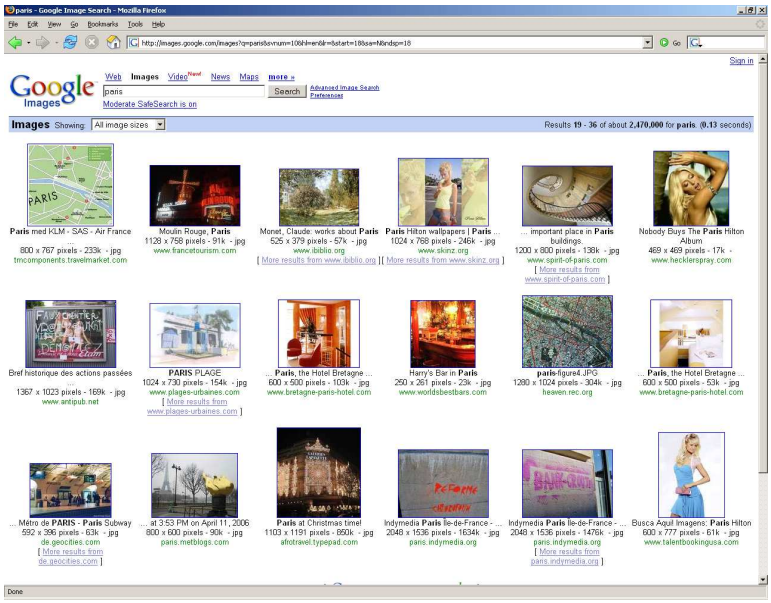


Figure 2.2: Search results for the keyword *paris* using Google Image Search.

Again, what we immediately notice is that the search engine fails to discriminate two categories of images: those related to the city of Paris and those showing Paris Hilton, the heiress to the Hilton fortune whose popularity on the Web could hardly be disputed.[1]

More striking is the quality of search results in general. While the search engine does a good job with retrieving documents, the results of image searches in general are disappointing. For the keyword *Paris* most of us would expect photos of places in Paris or maps of the city. In reality only about half of the photos on the first page, a quarter of the photos on the second page and a fifth on the third page are

[1] Although the most common search term of all times is "britney spears", see `http://www.google.com/press/zeitgeist.html`

directly related to our concept of Paris. The rest are about clouds, people, signs, diagrams etc.

The problem is that associating photos with keywords is a much more difficult task than simply looking for keywords in the texts of documents. Automatic image recognition is currently a largely unsolved research problem, which means that our computers cannot "see" what kind of object is on the photo. Search engines attempt to understand the meaning of the image solely from its context, e.g. based on the name of the file and the text that surrounds the image. Inevitably, this leads to rather poor results.

3. **Find new music that I (might) like**

   This query is at an even higher level of difficulty so much so that most of us wouldn't even think of posing it to a search engine. First, from the perspective of automation, music retrieval is just as problematic as image search. As in the previous case, a search engine could avoid the problem of understanding the content of music and look at the filename and the text of the web page for clues about the performer or the genre. We suspect that such search engines do not exist for different reasons: most music on the internet is shared illegally through peer-to-peer systems that are completely out of reach for search engines. Music is also a fast moving good; search engines typically index the Web once a month and therefore too slow for the fast moving world of music releases. (Google News, the news search engine of Google addresses this problem by indexing well-known news sources at a higher frequency than the rest of the Web.)

   But the reason we would not attempt to pose this query mostly has to do with formulating the music we like. Most likely we would search for the names of our favorite bands or music styles as a proxy, e.g. *"new release" ("Macy Gray" OR "Robbie Williams").* This formulation is awkward on the one hand because it forces us to query by example. It will not make it possible to find music that is similar to the music that we like but from different artists. In other words it will not lead us to discover new music.

   On the other hand, our musical taste might change in which case this query would need to change its form. A description of our musical taste is something that we might list on our homepage but it is not something that we would like to keep typing in again for accessing different music-related services on the internet. Ideally, we would like the search engine to take this information from our homepage or to grab it —with our permission— from some other service that is aware of our musical taste such as our online music store, internet radio stations we listen to or the music player of our own mp3 device.

4. **Tell me about music players with a capacity of at least 4GB.**

   This is a typical e-commerce query: we are looking for a product with certain characteristics.

   One of the immediate concerns is that translating this query from natural language to the boolean language of search engines is (almost) impossible. We could try the

search *"music player" "4GB"* but it is clear that the search engine will not know that 4GB is the capacity of the music player and we are interested in all players with at least that much memory (not just those that have exactly 4GB). Such a query would return only pages where these terms occur as they are. Problem is that general purpose search engines do not know anything about music players or their properties and how to compare such properties.

An even bigger problem is the one our machines face when trying to collect and aggregate product information from the Web. Again, a possibility would be to extract this information from the content of web pages. The information extraction methods used for this purpose have a very difficult task and it is easy to see why if we consider how a typical product description page looks like to the eyes of the computer. Even if an algorithm can determine that the page describes a music player, information about the product is very difficult to spot. We could teach the computer a heuristic that the price is the number that appears directly after the word "price". However, elements that appear close to each other on the page may not be close in the HTML source where text and styling instructions are mixed together. If we make the rules specific to a certain page our algorithm will not be able to locate the price on other pages or worse, extract the price of something else. (Price as a number is still among the easiest information to locate.)

Further, what one vendor calls "capacity" and another may call "memory". In order to compare music players from different shops we need to determine that these two properties are actually the same and we can directly compare their values.

In practice, information extraction is so unreliable that it is hardly used for product search. It appears in settings such as searching for publications on the Web. Google Scholar and CiteSeer are two of the most well-known examples. They suffer from the typical weaknesses of information extraction, e.g. when searching *York Sure*, the name of a Semantic Web researcher, Scholar returns also publications that are published in New York, but have otherwise nothing to do with the researcher in question. The cost of such errors is very low, however: most of us just ignore the incorrect results.

In the case of e-commerce search engines the cost of such mistakes is prohibitive. Information extraction-based engines such as Shopping.com improve accuracy by implementing specific rules of extraction for each website from which they extract product information. This method is limited by the human effort in locating websites and writing the rules of extraction. As a result, these comparison sites feature only a selected number of vendors.

Google's Froogle represents a different approach in that it relies on information provided by the shops themselves. This ensures the accuracy of information, because shop owners have access to the product database that is behind the web interface and they can transform the data in some format that is understood by the search engine. (In effect, they integrate their system to that of the search engine.) This method has a different drawback: shop owners need to provide information according a specific schema provided by the site owner. Such a schema prescribes

what kind of products there are and what properties they have. Information that does not fit the classification cannot submitted or searched upon. In effect, each of these sites limits the characteristics of products to the bare minimum (e.g. price and brand).

2.1.2 Diagnosis: A lack of knowledge

The questions above are arbitrary in their specificity but they illustrate a general problem in accessing the vast amounts of information on the Web. Namely, in all five cases we deal with a *knowledge gap*: what the computer understands and able to work with is much more limited than the knowledge of the user. The handicap of the computer is mostly due to technological difficulties in getting our computers to understand natural language or to "see" the content of images and other multimedia. Even if the information is there, and is blatantly obvious to a human reader, the computer may not be able to see anything else of it other than a string of characters. In that case it can still compare to the keywords provided by the user but without any understanding of what those keywords would mean.

This problem affects all of the above queries to some extent. A human can quickly skim the returned snippets (showing the context in which the keyword occurs) and realize that the different references to the word Harmelen do not all refer to persons and even the persons named Harmelen cannot all be the same. In the second query, it is also blatantly obvious for the human observer that not all pictures are of cities. However, even telling cities and celebrities apart is a difficult task when it comes to image recognition.

In most cases, however, the knowledge gap is due to the lack of some kind of *background knowledge* that only the human possesses. The background knowledge is often completely missing from the context of the Web page and thus our computers do not even stand a fair chance by working on the basis of the web page alone. In the case of the second query, an important piece of knowledge that the computer doesn't possess is the common knowledge that there is a city named Paris and there is a famous person named Paris Hilton (who is also different from the Hilton in Paris).

Answering the third query requires the kind of extensive background knowledge about musical styles, genres etc. that shop assistants and experts in music possess. This kind of knowledge is well beyond the information that is in the database of a typical music store. The third case is also interesting because there is also lacking background knowledge about the user. There has to be a way of providing this knowledge to the search engine in a way that it understands it.

The fourth query is noteworthy because it highlights the problem of aggregating information. The factual knowledge about particular products can be more or less extracted from the content of web pages, but if not, shop owners could be asked to provide it. It is unrealistic to expect, however, that all shops on the Web would agree to one unified product catalog (a listing of product types, properties, models etc) and provide information according to that schema. But if each shop provides information using its own classification we need additional knowledge in order to merge data from different catalogs. For example, we need to know that "mp3 players" and "mp3 capable mobile phones" both fit

the category of digital music players, that “capacity” and “memory” are the same things and that 500 dollars is the equivalent of (about) 400 euros.

2.2 The semantic solution

The idea of the Semantic Web is to apply advanced knowledge technologies in order to fill the knowledge gap between human and machine. This means providing knowledge in forms that computers can readily process and reason with. This knowledge can either be information that is already described in the content of the Web pages but difficult to extract or additional background knowledge that can help to answer queries in some way. In the following we describe the improvement one could expect in case of our four queries based on examples of existing tools and applications that have been implemented for specific domains or organizational settings.

In the case of the first query the situation can be greatly improved by providing personal information in a semantic format. Although we will only cover the technological details in Chapter 5 and 6, an existing solution is to attach a semantic profile to personal web pages that describe the same information that appears in the text of the web page but in a machine processable format. The Friend-of-a-Friend (FOAF) project provides a widely accepted vocabulary for such descriptions. FOAF profiles listing attributes such as the name, address, interests of the user can be linked to the web page or even encoded in the text of the page. As we will see several profiles may also exist on the Web describing the same person. As all profiles are readable and comparable by machines, all knowledge about a person can be combined automatically.

For example, Frank van Harmelen has such a profile attached to his homepage on the Web. This allows a search engine to determine that the page in question is about a person with specific attributes. (Thus pages about persons and villages would not be confused.) Assuming that all other van Harmelens on the Web would provide similar information, the confusion among them could also be easily avoided. In particular, the search engine could alert us to the ambiguity of our question and ask for some extra information about the person we are looking for. The discussion with the search engine would be very different:

> Q: *Who is Frank van Harmelen?*
>
> A: *Your question is ambiguous: there is a great deal of information about a Frank van Harmelen who is a professor at the Vrije Universiteit. However, there are other persons named Harmelen and also a village in the municipality of Woerden. Which one did you mean?*

Similarly, the solution in the second case is to attach metadata to the images in question. For example, the online photo sharing site Flickr allows to annotate images using geographic coordinates. After uploading some photos users can add keywords to describe their images (e.g. “Paris, Eiffel-tower”) and drag and drop the images on a geographic map to indicate the location where the photo was taken. In the background the system computes the latitude and longitude of the place where the user pointed and attaches this

information to the image. Searching of photos of Paris becomes a breeze: we can look up Paris on the map and see what other photos have been put there by other users. Although in this case the system is not even aware that Paris is a city, a minimal additional information about photos (the geo-coordinates) enables a kind of visualization that makes the searching task much easier. And if over time the system notes that most images with the keyword “Paris” fall in a specific geographic area on the map, it can even conclude that Paris is a place on the map (see Figure 2.3).

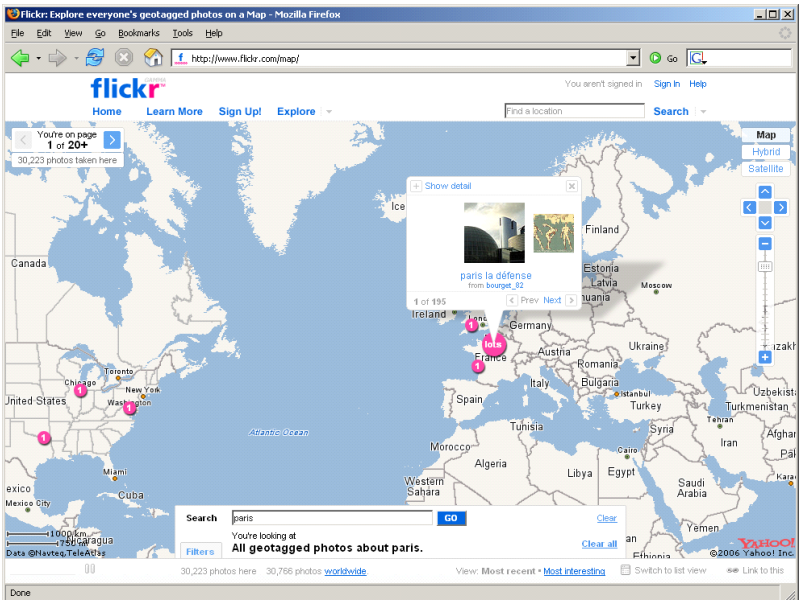


Figure 2.3: Searching for the keyword Paris using the geographic search of Flickr.

The same technique of annotating images with metadata is used in the MultiMediaN research project to create a joint catalogue of artworks housed in different collections.[2] The knowledge provided extends to the content of images as well as other attributes such as the creator, style or material of the work. When these annotations are combined with existing background knowledge about the artists, art periods, styles etc., it becomes possible to aggregate information and to search all collections simultaneously while avoiding the pitfalls of keyword based search. For example, when searching for “Art Nouveau” the system not only returns images that are explicitly annotated as Art Nouveau in style, but also other works by painters who have been known to belong to the Art Nouveau movement. Further, it is aware that the term “palingstijl” is the Dutch equivalent of Art Nouveau and therefore works annotated with this term are also relevant to the query.

[2] `http://e-culture.multimedian.nl/demo/search`

The background knowledge required for recommending music is already at work behind the online radio called Pandora[3]. Pandora is based on the Music Genome Project, an attempt to create a vocabulary to describe characteristics of music from melody, harmony and rhythm, to instrumentation, orchestration, arrangement, lyrics, and the rich world of singing and vocal harmony. Over several years thousands of songs have been annotated by experts in music theory. This knowledge is now used by the system to recommend unknown music to users based on their existing favorites. The Foafing the Music project combines this idea with the retrieval of information about music releases and upcoming events related to our favorite artists.[4] A particularly interesting feature of this system is that it can reuse information about our musical tastes from other sources such as a personal profile on one's homepage or an online playlist of our recently played music. In order to track the fast-paced world of music, the system tracks newsfeeds updated by data providers on a daily basis.

Our fourth problem, the aggregation of product catalogs can also be directly addressed using semantic technology. As we have seen the problem in this case is the difficulty of maintaining a unified catalog while keeping the classifications flexible (e.g. new categories) and without requiring an exclusive commitment to the catalogue. This is the problem that Vodafone faced when putting together its Vodafone Live! portal, which is a catalog of content provided by partners of Vodafone [Appelquist, 2005]. The semantic solution to this problem is to capture the commonalities of mobile content in a single shared schema that all partners follow (e.g. the major categories such as Games, Chat, Erotica etc.), while giving content providers the flexibility to extend this schema as they see fit. Using semantic technology page views per download decreased 50% at Vodafone while ringtone sales went up 20% in just two months. However, there are also advantages for the content providers: if another operator (say, T-Mobile) would ask them to classify their content according to a different schema all that would be required is a mapping between the classifications used by the two companies.

2.3 Key concepts of the Semantic Web

The idea of the Semantic Web is to extend unstructured information with machine processable descriptions of the meaning (semantics) of information and to provide missing background knowledge where required.

The key challenge of the Semantic Web is to ensure the interoperability of knowledge. Ontologies and ontology languages are the key enabling technology in this respect. An ontology, by its most cited definition in AI[5], is a shared, formal conceptualization of a domain [Gruber, 1993, Borst et al., 1997]. Ontologies are data models with two special characteristics, which lead to the notion of shared meaning or semantics:

[3] `http://www.pandora.com`

[4] `http://foafing-the-music.iua.upf.edu/`

[5] The term ontology originates from Philosophy. Ontology (with a capital O) is the most fundamental branch of metaphysics. It studies being or existence as well as the basic categories thereof - trying to find out what entities and what types of entities exist. See `http://en.wikipedia.org/wiki/Ontology`

1. Ontologies build upon a shared understanding within a *community*. This understanding represents an agreement of members of the community over the concepts and relationships that are present in a domain.

2. Ontologies can be expressed in formal languages that are specifically designed for a web-based scenario and have a well-defined logic-based semantics. RDF [Manola and Miller, 2004] and OWL [McGuinness and van Harmelen, 2004]) have standard syntaxes and logic-based formal semantics, which allows computers to manipulate them, including reasoning with or with the help of ontologies. We will return to these language in Chapter 5.

The role of ontologies and semantics can be explained with an analogy from the world of libraries. The current Web resembles an old library filled with books but without a library catalogue. Librarians search this library by looking through all books in search for the exact words that the visitor of the library mentioned. This method is the basis of current search engines such as Google, even if their precision is significantly boosted by additional techniques such as considering the linkage of web pages.

In comparison, a modern library provides a computerized catalogue to its patrons. This catalogue contains a record about each item in the library with meta-information such as the author, title and publisher of the book, but also the classification of the book in terms of disciplines and interests. Such a system allows the librarian to search the collection by looking through the catalogue instead of the books and makes it possible to search by a particular field (such as the year of publication) and by meta-information that may not even be explicitly mentioned in the item (such as the classification of the topic or the rating of a movie).

Most importantly, the syntax and semantics of such records —the fields of the record and the interpretation of the values— are laid down in international standards such as the MARC format[6]. The vocabularies used for classifying the content of books are defined using taxonomies (concept hierarchies) such as Dewey Decimal Classification System[7] (DDC) used in the US or the newer Universal Decimal Classification System[8] (UDC), which is in use in Europe.

These standards are important enablers for the electronic exchange of information: while libraries around the world may use different information systems for managing their collections, these systems can all exchange information due to the standard syntax of MARC records and the shared interpretation of classifications. This means that library records need to be authored only once (for example by the publisher of a new book) and can then be shared within all libraries that share the same set of standards.

Similarly for the Web, ontologies (such as MARC, the DDC or the UDC) capture agreements that are required to make sure that the fields and the values provided as metadata are interpreted uniformly within a community. In case where several standards exist (such as for the classification of topics) mappings need to be established before the information can be shared. The role of Semantic Web technology is then to support —and

[6] `http://www.loc.gov/marc/`
[7] `http://en.wikipedia.org/wiki/Dewey_Decimal_Classification`
[8] `http://www.udcc.org/`

where possible automate— the process of creating ontologies and metadata either from scratch or from existing source, storing and reasoning with ontologies, the finding of correct mappings between ontologies and facilitating the development of ontology-based end-user applications.

The Semantic Web was originally conceptualized as an extension of the current Web, i.e. as the application of metadata for describing Web content. In this vision, the content that is already on the Web (text, but also multimedia) would be enriched in a collaborative effort by the users of the Web. However, this vision was soon considered to be less realistic as their was a widespread belief that Semantic Web technologies would be too difficult to master for the average person contributing content to the Web. (This was before the successes of Web 2.0 demonstrated the effectiveness of harnessing the power of users for collaborative annotation, see Section 2.5). The alternative view predicted that the Semantic Web will first break through behind the scenes and not with the ordinary users, but among large providers of data and services. The second vision predicts that the Semantic Web will be primarily a "web of data" operated by data and service providers largely unbeknown to the average user.

That the Semantic Web is formulated as a vision points to the problem of bootstrapping the Semantic Web. As many modern technologies, the Semantic Web suffers from what the economist Kevin Kelly calls the *fax-effect*[9]. Kelly notes that when the first fax machines were introduced, they came with a very hefty price tag. Yet they were almost useless: namely, the usefulness of a fax comes from being able to communicate with other fax users. In this sense every fax unit sold increases the value of all fax machines in use. While traditional goods such as the land or precious metals become more valuable the less is produced (called the law of scarcity), the fax machine network exhibits the opposite, which is called the law of plentitude.[10]

So is it with the Semantic Web: at the beginning the price of technological investment is very high. One has to adapt the new technology which requires an investment in learning. Further, the technology needs time to become reliable. But most of all there need to be other adopters who also provide data in semantic formats and follow the basic protocols of interoperability such as the SPARQL query language and protocol for accessing remote data stores [Prud'hommeaux and Seaborne, 2006, Clark, 2006]. The reader may note that this is nothing new: the first Web had to deal with the exact same effect. In fact, reading the account of Tim Berners-Lee one cannot help but wonder at the extent to which making the Web successful depended on Berners-Lee charisma in getting the first users on board among a general disbelief [Berners-Lee et al., 1999].

What makes the case of the Semantic Web more difficult, however, is an additional cost factor. Returning to the example of the fax network, we can say that it required a certain kind of agreement to get the system working on a global scale: all fax machines needed to adopt the same protocol for communicating over the telephone line. This is similar to the case of the Web where global interoperability is guaranteed by the standard protocol for communication (HTTP). However, in order to exchange meaning there has

[9] `http://www.kk.org/newrules/newrules-3.html`

[10] Metcalfe's law is an attempt to quantify utility in networked systems. It states that the value of a telecommunications network is proportional to the square of the number of users of the system (n^2). See `http://en.wikipedia.org/wiki/Metcalfe's_law` for the discussion.

to be a minimal external agreement on the meaning of some primitive symbols, i.e. on what is communicated through the network. It is as if when installing the fax machine we would have to call our partners in advance to agree on the meaning of some of the symbols that we will use in our fax communications.

This agreement doesn't have to be complete as the meaning of unknown symbols can often be deduced from the context, in particular the meaning of related symbols and the meaning of relationships between symbols. Often a little semantics is enough to solve important tasks, hence the mantra "a little semantics goes a long way". Our machines can also help in this task to the extent that some of the meaning can be described in formal rules (e.g. if A is true, B should follow). But formal knowledge typically captures only the smaller part of the intended meaning and thus there needs to be a common grounding in an external reality that is shared by those at separate ends of the line.

2.4 Development of the Semantic Web

The vision of extending the current human-focused Web with machine processable descriptions of web content has been first formulated in 1996 by Tim Berners-Lee, the original inventor of the Web [Berners-Lee et al., 1999]. The Semantic Web has been actively promoted since by the World Wide Web Consortium (also led by Berners-Lee), the organization that is chiefly responsible for setting technical standards on the Web. As a result of this initial impetus and the expected benefits of a more intelligent Web, the Semantic Web has quickly attracted significant interest from funding agencies on both sides of the Atlantic, reshaping much of the AI research agenda in a relatively short period of time[11]. In particular, the field of Knowledge Representation and Reasoning took center stage, but outcomes from other fields of AI have also been put into to use to support the move towards the Semantic Web: for example, Natural Language Processing and Information Retrieval have been applied to acquiring knowledge from the World Wide Web.

As the Semantic Web is a relatively new, dynamic field of investigation, it is difficult to precisely delineate the boundaries of this network.[12] For our research on the Semantic Web community we have defined the community by including those researchers who have submitted publications or held an organizing role at any of the past International Semantic Web Conferences (ISWC02, ISWC03, ISWC04) or the Semantic Web Working Symposium of 2001 (SWWS01), the most significant conference series devoted entirely to the Semantic Web.[13] We note that another commonly encountered way of defining the boundary of a scientific community is to look at the authorship of representative journals

[11] Examples of some of the more significant projects in the area include the US DAML program funded by DARPA and a number of large projects funded under the IST initiative of the EU Fifth Framework Programme (1998-2002) and the Strategic Objective 2.4.7 of the EU Sixth Framework Programme (2002-2006).

[12] In fact, it is difficult to determine at what point does a new research concept become a separate field of investigation. With regard to Semantic Web, it is clear that many of the scientists involved have developed ties before their work on the Semantic Web, just as some of the research published in the Semantic Web area has been worked out before in different settings.

[13] Besides the international conferences, there have European Semantic Web Conferences (ESWC) held since 2004 and the first Asian Semantic Web Conference (ASWC) will be held in 2006.

(see e.g. [Heimeriks et al., 2003]). However, the Semantic Web hasn't had a dedicated journal until 2004 and still most Semantic Web related publications appear in AI journals not entirely devoted to the Semantic Web.

The complete list of individuals in this community consists of 608 researchers mostly from academia (79%) and to a lesser degree from industry (21%). Geographically, the community covers much of the United States, Europe, with some activity in Japan and Australia (see Figure 2.4). As Figure 2.5 shows, the participation rate at the individual ISWC events have quickly reached the level typical of large, established conferences and remained at that level even for the last year of data (2004), when the conference was organized in Hiroshima, Japan. The number of publications written by the members of the community that contain the keyword "Semantic Web" has been sharply rising since the beginning.

Figure 2.4: Semantic Web researchers and their network visualized according to geography.

While the research effort behind the Semantic Web is immense and growing dynamically, Semantic Web technology has yet to see mainstream use on the Web and in the enterprise. In the following, we will illustrate the growth of the Semantic Web by tracing its popularity on the Web. Although this method does not allow us to survey the use of Semantic Web technology in enterprise or governmental settings, we believe that internal applications of Semantic Web technology ("Semantic Webs" within organizations) are likely lagging behind due to the natural caution with which industry and the government treat any new technology.

To follow the popularity of Semantic Web related concepts and Semantic Web standards on the Web, we have executed a set of temporal queries using the search engine

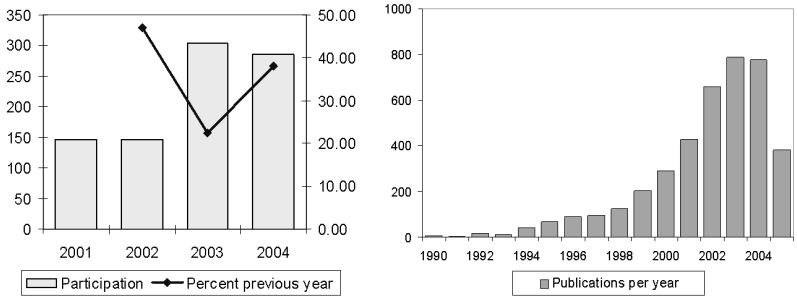


Figure 2.5: Participation at the international Semantic Web events (2001-2004) and publications per year (1990-2005).

Altavista. The queries contained single terms plus a disambiguation term where it was necessary. Each query measured the number of documents with the given term(s) at the given point in time.

Figure 2.6 shows the number of documents with the terms *basketball*, *Computer Science*, and *XML*. We have divided all counts with the number of documents with the word *web* to account for the general growth of the Web. The flat curve for the term basketball validates this strategy: we could have expected the popularity of basketball to be roughly stable over this time period. Computer Science takes less and less share of the Web as the Web shifts from scientific use to everyday use. The share of XML, a popular pre-semantic web technology seems to grow and stabilize as it becomes a regular part of the toolkit of Web developers.

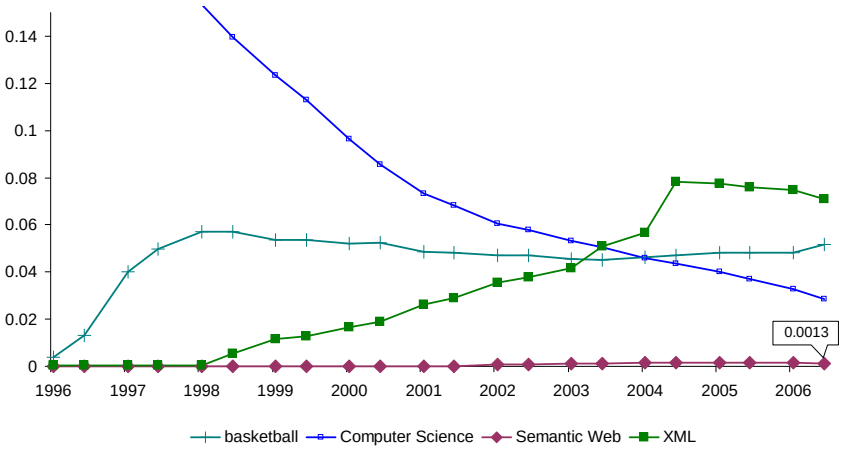


Figure 2.6: Number of webpages with the terms *basketball*, *Computer Science*, and *XML* over time and as a fraction of the number of pages with the term *web*.

Against this general backdrop we can also look at the share of Semantic Web related terms and formats, in particular the terms *RDF*, *OWL* and the number of ontologies (Semantic Web Documents) in RDF or OWL. As Figure 2.7 shows most of the curves have flattened out after January, 2004. It is not known at this point whether the dip in the share of Semantic Web is significant. While the use of RDF has settled at a relatively high level, OWL has yet to break out from a very low trajectory (only 4020 OWL documents were found by the search engine in June, 2006).

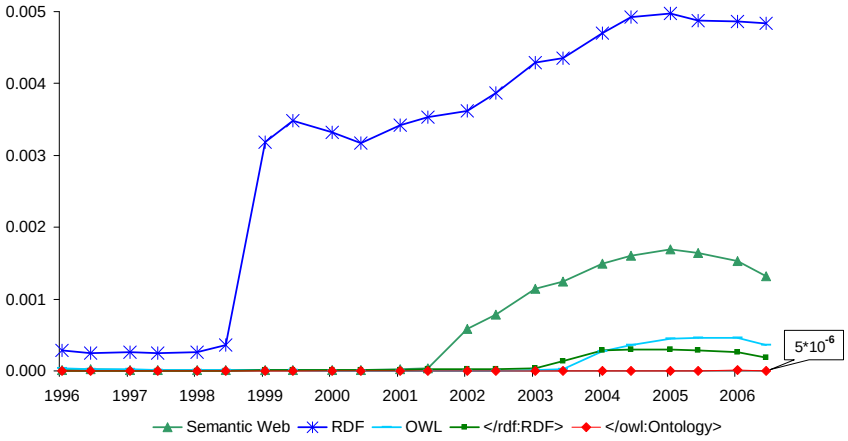


Figure 2.7: Number of webpages with the terms *RDF*, *OWL* and the number of ontologies (Semantic Web Documents) in RDF or OWL over time. Again, the number is relative to the number of pages with the term *web*.

It is also interesting to look at the share of the mentioning of Semantic Web formats versus the actual number of Semantic Web documents using that format. The resulting *talking vs. doing* curve shows the phenomenon of technology hype in both the case of XML, RDF and OWL (see Figure 2.8). The hype reaches its maximum following the release of the corresponding W3C specifications: this is the point where the technology "makes the press" and after which its becoming increasingly used on the Web.[14] Interestingly, the XML and RDF hypes both settle at a fix value: there are roughly 15 pages mentioning XML for every XML document and 10 pages mentioning RDF for every RDF document. OWL has not yet settled at such a value but seems to be approaching it.

It is interesting to note that these curves are very similar in shape to the well known five-stage *hype cycle* of Gartner Research.[15] The hype cycle is defined by Gartner as follows:

[14] XML 1.0 was released as a W3C specification on February 10, 1998, see `http://www.w3.org/TR/1998/REC-xml-19980210`. RDF first became a W3C recommendation on February 22, 1999, see `http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/`. OWL became a recommendation on February 10, 2004.

[15] `http://www.gartner.com/pages/story.php.id.8795.s.8.jsp`

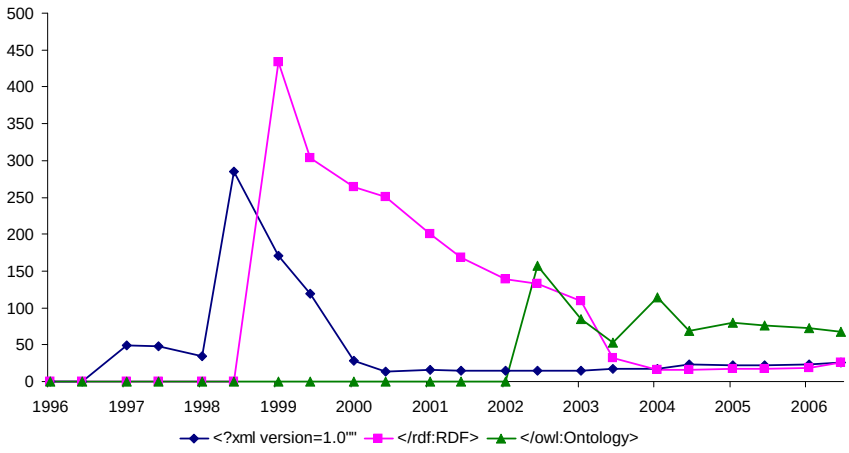


Figure 2.8: The hype cycle of Semantic Web related technologies as shown by the number of web pages about a given technology relative to its usage.

> The first phase of a Hype Cycle is the "technology trigger" or breakthrough, product launch or other event that generates significant press and interest. In the next phase, a frenzy of publicity typically generates over-enthusiasm and unrealistic expectations. There may be some successful applications of a technology, but there are typically more failures. Technologies enter the "trough of disillusionment" because they fail to meet expectations and quickly become unfashionable. Consequently, the press usually abandons the topic and the technology. Although the press may have stopped covering the technology, some businesses continue through the "slope of enlightenment" and experiment to understand the benefits and practical application of the technology. A technology reaches the "plateau of productivity" as the benefits of it become widely demonstrated and accepted. The technology becomes increasingly stable and evolves in second and third generations. The final height of the plateau varies according to whether the technology is broadly applicable or benefits only a niche market.

Although the word hype has attracted some negative connotations, we note that hype is unavoidable for the adoption of network technologies such as the Semantic Web that exhibit the fax-effect described in Section 2.3. Namely creating technology visions is unavoidable in bootstrapping technology adoption as there are relative meager immediate benefits to early adopters compared to the value that a complete network could bring. The disillusionment that follows is also unavoidable partly because the technology will need to mature but also because the technology will prove to be useful in different ways than originally predicted.

It is not surprising then that the adoption of the Semantic Web is also taking different paths than originally laid out in the often quoted vision of Tim Berners-Lee, which

appeared on the pages of Scientific American in 2001 [Berners-Lee et al., 2001]. While standardization of the Semantic Web is mostly complete, Semantic Web technology is not reaching yet the mainstream user and developer community of the Web. In particular, the adoption of RDF is lagging behind XML, even though it provides a better alternative and thus many hoped it would replace XML over time. (Note that some of the benefits are in fact time-bound. Flexibility — a key benefit of a semantic infrastructure — is expected to show over time in easier adaptation of applications for serving different tasks based on different combinations of existing data and services.)

On the other hand, the recent support for Semantic Web standards by vendors such as Oracle[16] will certainly inspire even more confidence in the corporate world, leading to an adoption of semantic technologies for data and service integration within the enterprise. This could lead an earlier realization of the vision of the Semantic Web as a "web of data", which could ultimately result in a resurgence of general interest on the Web.

2.5 The emergence of the social web

Hyperlinked as it may have been, the early web of the mid-1990s has been largely a place comparable to a bulletin board to which each passer by would affix some basic personal information or a shiny corporate catalog. Although Tim Berners-Lee envisioned a read/write Web (the very first browser also worked as an HTML editor), the Web was a read-only medium for a majority of users. This passive attitude toward the Web was broken by a series of changes in usage patterns and technology that are now referred to as Web 2.0. The changes that led to the current level of social engagement online may not have been radical or significant individually, which explains why the term Web 2.0 has been created largely after the fact to represent the evolution of the Web. Nevertheless, these set of innovations in the architecture and usage patterns of the Web led to an entirely different role of the online world as a platform for socialization. The resulting increase in our capacity to obtain information and social support online can be quantified: a recent major survey based on interviews with 2200 adults shows that the internet significantly improves Americans' capacity to maintain their social networks despite early fears about the effects of diminishing real life contact. The survey confirms that not only networks are maintained and extended online, but they are also successfully activated for dealing with major life situations such as getting support in case of a major illness, looking for jobs, informing about major investments etc. [Boase et al., 2006]

The first wave of socialization on the Web was due to the appearance of weblogs, wikis and other forms of web-based communication and collaboration. Blogs and wikis attracted mass popularity from around 2003 (see Figure 2.9). What they have in common is that they both significantly lower the requirements for adding content to the Web: editing blogs and wikis does not require any knowledge of HTML. Blogs and wikis allowed individuals and groups to claim their personal space on the Web and fill it with content at relative ease.

While weblogs have been first assessed as purely personal publishing (similar to diaries), nowadays the blogosphere is widely recognized as a densely interconnected net-

[16] See `http://www.oracle.com/technology/tech/semantic_technologies/index.html`

work through which news, ideas and influences travel rapidly as bloggers reference and reflect on each other's postings.

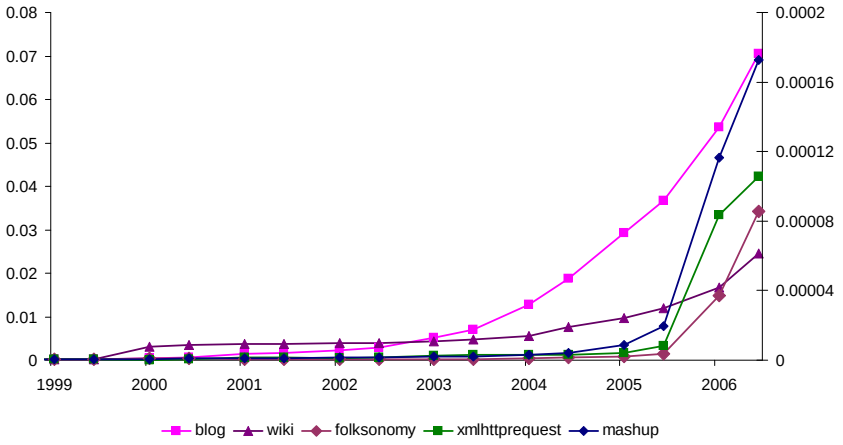


Figure 2.9: Development of the social web. The fraction of webpages with the terms *blogs*, *wiki* over time is measured on the left vertical axis. The fraction of webpages with the terms *folksonomy*, *XmlHttpRequest* and *mashup* is measured on the right hand vertical axis.

Although the example of Wikipedia is outstanding, wikis large and small are used by groups of various sizes as an effective knowledge management tool for keeping records, describing best practices or jointly developing ideas. Regardless of the goal, the collective ownership of a Wiki at best enforces a sense of community or at the worst reveals a lack of cohesion in social groups. Similarly, the significance of instant messaging (ICQ) is also not just instant communication (phone is instantaneous, and email is almost instantaneous), but the ability to see who is online, a transparency that induces a sense of social responsibility.

Social networking services entered the field at the same time as blogging and wikis started to take off. The first-mover Friendster[17] attracted over 5 million registered users in the span of a few months [Kahney, 2003] in 2003, which was followed by Google and Microsoft starting or announcing similar services. Although these sites feature much of the same content that appear on personal web pages, they provide a central point of access and bring structure in the process of personal information sharing and online socialization. Following registration, these sites allow users to post a profile with basic information, to invite others to register and to link to the profiles of their friends. The system also makes it possible to visualize and browse the resulting network in order to discover friends in common, friends thought to be lost or potential new friendships based on shared interests. (Thematic sites cater to more specific goals such as establishing a business contact or finding a romantic relationship.)

[17] `http://www.friendster.com`

These vastly popular systems allow users to maintain large networks of personal and business contacts. Members have soon discovered, however, that networking is only a means to an end in the cyberspace as well. The latest services are thus using user profiles and networks to stimulate different exchanges: photos are shared in Flickr, bookmarks are exchanged in del.icio.us, plans and goals unite members at 43Things. The idea of network based exchange is based on the sociological observation that social interaction creates similarity and vice versa, interaction creates similarity: friends are likely to have acquired or develop similar interests. As we will see in Chapter 10 many of these systems build on collaborative annotation (*folksonomies*) to connect users to relevant content and others interested in similar things. Much like Wikis, the new breed of websites achieve engagement by giving an active role to their community of users in creating and managing content. Explicit user profiles make it possible for these systems to introduce rating mechanism whereby either the users or their contributions are ranked according to usefulness or trustworthiness. Ratings are explicit forms of social capital that regulate exchanges in online communities in much the same way that reputation moderates exchanges in the real world.

The technological underpinnings of Web applications have also evolved in order to make the user experience of interacting with the Web as smooth as possible. This technological revolution in the conceptualization, design and implementation of web sites is referred to as Web 2.0, a buzzword coined by Tim O'Reilly.[18] The term Web 2.0 expresses the fact that the web has reached a new stage of development the same gradual way that software versions follow each other.

Web 2.0 is difficult to define solely on the basis of technological development as there have been no changes in the Web architecture required and much of the technologies at the hand of developers are hardly new at all.[19] What can be observed is a preference for formats, languages and protocols that are easy to use and develop with (in particular script languages, formats such as JSON, protocols such as REST) and thus support rapid development and prototyping. (Flickr, for example, is known to adapt the user interface several times a day.) For similar reasons, the new websites put the emphasis on accessibility, responsiveness and attractive, but minimalist design. Borrowing much of the ideology of the open source software movement they also open up their data and services for user experimentation: Google, Yahoo and countless smaller web sites expose key features of their systems through lightweight APIs while content providers do the same with information in the form of RSS feeds. The results of user experimentation with combinations of technologies are the so-called mashups, websites based on combinations of data and services provided by others. The best example of this development are the mashups based on Google's mapping service such as HousingMaps[20].

Web 2.0 is often contrasted to the Semantic Web, which is a more conscious and carefully orchestrated effort on the side of the W3C to trigger a new stage of developments

[18]Tim O'Reilly is founder and CEO of O'Reilly Media, a technology publisher. Tim O'Reilly's original article on Web 2.0 can be found at `http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html`

[19]This is certainly true for AJAX (Asynchronous JavaScript and XML) which drives many of the latest websites. AJAX is merely a mix of technologies that have been supported by browsers for years.

[20]See `http://housingmaps.org`

using semantic technologies. In practice the ideas of Web 2.0 and the Semantic Web are not exclusive alternatives: while Web 2.0 mostly effects how the front-end of websites are conceptualized, designed and implemented, the Semantic Web opens new technological opportunities for web developers in combining data and services from different sources. In particular, creating mashups becomes significantly easier with semantic technology.

Today's web is thus a place where social networks are explicitly described, maintained and put to work in acts of social exchange (both in form of information sharing and support) and creativity. The boundaries between creators and users are blurring thanks to technology that significantly lowers the barrier for developing new content and services. Technology also fosters authorship by enabling and promoting the sharing and reuse of building blocks (pieces of text, data, images, software) created by others. Issues of trust, privacy and ethics are dealt with through the increased social transparency provided by these new systems and services.

2.6 Discussion

The Semantic Web first appeared ten years ago as the ambitious vision of a future Web where all content is described in a way that computers can easily access, including the information that is currently only described in human language, captured by images, audio and video, or locked up in the many databases behind complex websites. Using such a web as an information source we would be able to automate complex tasks by automatically finding and pulling together the relevant pieces of information, adding the necessary background knowledge and applying human-like logic to decide complex questions, create plans, designs, find patterns and anomalies etc. In this Chapter we have only introduced the key idea behind the Semantic Web; technological details will follow in Chapters 5 and 6.

Although the Web has been already naturally evolving toward a separation of content and representation, the vision of the Semantic Web is rather revolutionary: not only major technological changes are required to realize this vision, but also a social adoption process had to be bootstrapped. In terms of technology, the past years have seen the formation of a substantial Semantic Web research community that delivers the innovative solutions required to realize the Semantic Web. (We will investigate this community in more detail in Chapter 9.) The World Wide Web Consortium (W3C) as the main body of standardization in the area of web technologies has also proceeded to issue official recommendations about the key languages and protocols that would guarantee interoperability on the Semantic Web.

The W3C is also the main propelling force behind the social adoption of the Semantic Web. We have tracked this process by looking at the mentioning and usage of certain key technological terms on the Web. We have noted that the social adoption of Semantic Web technology is lagging behind the scientific and standardization efforts, but it has the potential to take off in an exponential fashion due to the network effect. (The more data and services become available in Semantic Web compatible formats, the more attractive it becomes for others to switch to publish their data and services using the new technology.) The necessary push may come from large companies or governments where the

complexity of data and service integration tasks could justify an investment in Semantic Web technologies based on internal benefits alone.

Contrary to the development of the Semantic Web, the notion of a Web 2.0 has emerged from the developer community of the Web as a characterization of incremental technological developments that have collective led to a distinct class of web applications. These applications focus on the social aspects of the Web and attempt to harness the power of a community of users in building up and organizing valuable information. The role of technology in this effort is relatively minor: the technological changes required focus on making attractive and efficient interfaces for manipulating content.

Semantic Web and Web 2.0 are likely to benefit from a closer interaction in the future. Among others, the creation of mashups or combinations of diverse sources of data and services could greatly benefit from the shared representations and protocols proposed by the Semantic Web community. In Chapter 10, we will return to see how the Semantic Web itself could learn from the study of folksonomies, the bottom-up, user generated semantic structures that are used for organizing information in many Web 2.0 applications. We will find that folksonomies bring into play a so far unexplored dimension of knowledge representation, namely the social networks of the user community that creates annotations.

Chapter 3

Social Network Analysis

How do we rank pages on the Web? How does HIV spread? How do we explain the success or failure of entrepreneurs in terms of their business contacts? What is the advantage of terrorist networks built from loosely coupled cells?

All these questions have in common that they can be rephrased using the vocabulary of network analysis, a branch of sociology and mathematics that is increasingly applied also to questions outside the social domain. In the following we give an introduction to the main theory and methods of Social Network Analysis, which will be applied later to the analysis of social networks (Section 9) as well as semantic networks (Section 10). By no means do we expect to provide a complete coverage of any topic involved. For a more encyclopedic treatment of network analysis we refer the reader to the social network analysis reference of Wasserman and Faust [Wasserman et al., 1994].

While Social Science is often looked upon by researchers from the exact sciences as vague and thus necessarily inconclusive, network analysis should appeal to all as one of the most formalized branches of Social Science. Most of these formalisms are based on the simple nodes and edges representations of social networks to which a large array of measures and statistics can be applied. While some of the more sophisticated of these methods require a deep mathematical understanding to be applied correctly, the simple concepts discussed in this Chapter should be easily understood by anyone with an advanced level of secondary-school mathematics.

3.1 What is network analysis?

Social Network Analysis (SNA) is the study of social relations among a set of actors. The key difference between network analysis and other approaches to social science is the focus on relationships between actors rather than the attributes of individual actors. Network analysis takes a global view on social structures based on the belief that types and patterns of relationships emerge from individual connectivity and that the presence (or absence) of such types and patterns have substantial effects on the network and its constituents. In particular, the network structure provides opportunities and imposes con-

straints on the individual actors by determining the transfer or flow of resources (material or immaterial) across the network.

The focus on relationships as opposed to actors can be easily understood by an example. When trying to predict the performance of individuals in a scientific community by some measure (say, number of publications), a traditional social science approach would dictate to look at the attributes of the researchers such as the amount of grants they attract, their age, the size of the team they belong to etc. A statistical analysis would then proceed by trying to relate these attributes to the outcome variable, i.e. the number of publications.

In the same context, a network analysis study would focus on the interdependencies within the research community. For example, one would look at the patterns of relationships that scientists have and the potential benefits or constraints such relationships may impose on their work. For example, one may hypothesize that certain kinds of relationships arranged in a certain pattern may be beneficial to performance compared to the case when that pattern is not present. The patterns of relationships may not only be used to explain individual performance but also to hypothesize their impact on the network itself (network evolution). Attributes typically play a secondary role in network studies as control variables.[1]

SNA is thus a different approach to social phenomena and therefore requires a new set of concepts and new methods for data collection and analysis. Network analysis provides a vocabulary for describing social structures, provides formal models that capture the common properties of all (social) networks and a set of methods applicable to the analysis of networks in general. The concepts and methods of network analysis are grounded in a formal description of networks as graphs. Methods of analysis primarily originate from graph theory as these are applied to the graph representation of social network data. (Network analysis also applies statistical and probabilistic methods and to a lesser extent algebraic techniques.)

It is interesting to note that the formalization of network analysis has brought much of the same advantages that the formalization of knowledge on the Web (the Semantic Web) is expected to bring to many application domains. Previously vaguely defined concepts such as *social role* or *social group* could now be defined on a formal model of networks, allowing to carry out more precise discussions in the literature and to compare results across studies.

The methods of data collection in network analysis are aimed at collecting relational data in a reliable manner. Data collection is typically carried out using standard questionnaires and observation techniques that aim to ensure the correctness and completeness of network data. Often records of social interaction (publication databases, meeting notes, newspaper articles, documents and databases of different sorts) are used to build a model of social networks. We return to the particular use of electronic data (data from the Internet and the Web) in Chapter 4.

[1] The role of control variables in statistical analysis is to exclude the effect of non-network variables on the outcome variable.

3.2 Development of Social Network Analysis

The field of Social Network Analysis today is the result of the convergence of several streams of applied research in sociology, social psychology and anthropology.

Many of the concepts of network analysis have been developed independently by various researchers often through empirical studies of various social settings. For example, many social psychologists of the 1940s found a formal description of social groups useful in depicting communication channels in the group when trying to explain processes of group communication. Already in the mid-1950s anthropologists have found network representations useful in generalizing actual field observations, for example when comparing the level of reciprocity in marriage and other social exchanges across different cultures.

Some of the concepts of network analysis have come naturally from social studies. In an influential early study at the Hawthorne works in Chicago, researchers from Harvard looked at the workgroup behavior (e.g. communication, friendships, helping, controversy) at a specific part of the factory, the bank wiring room [Mayo, 1933]. The investigators noticed that workers themselves used specific terms to describe who is in "our group". The researchers tried to understand how such terms arise by reproducing in a visual way the group structure of the organization as it emerged from the individual relationships of the factory workers (see Figure 3.1).[2] In another study of mixed-race city in the Southern US researchers looked at the network of overlapping "cliques" defined by race and age [Warner and Lunt, 1941].[3] They also went further than the Hawthorne study in generating hypotheses about the possible connections between cliques. (For example, they noted that lower-class members of a clique are usually only able to connect to higher-class members of another clique through the higher-class members of their own clique.)

Despite the various efforts, each of the early studies used a different set of concepts and different methods of representation and analysis of social networks. However, from the 1950s network analysis began to converge around the unique world view that distinguishes network analysis from other approaches to sociological research. (The term "social network" has been introduced by Barnes in 1954.) This convergence was facilitated by the adoption of a graph representation of social networks usually credited to Moreno. What Moreno called a *sociogram* was a visual representation of social networks as a set of nodes connected by directed links. The nodes represented individuals in Moreno's work, while the edges stood for personal relations. However, similar representations can be used to depict a set of relationships between any kind of social unit such as groups, organizations, nations etc. While 2D and 3D visual modelling is still an

[2]The study became famous not so much of the network methods used but for what became known in management science as the *Hawthorne-effect*. In brief, managers at the Hawthorne factory were initially trying to understand what alterations in the work conditions affect productivity. To their surprise no matter what the change was it seemed to affect productivity in a positive way. Mayo and colleagues concluded that the mere participation in the research project itself was the key factor as workers were pleased with the management taking an interest in their conditions. Although it became widely known, the original study as well as the general existence of this effect is disputed [Gillespie, 1993].

[3]Clique is a term that now has a precise definition in network analysis.

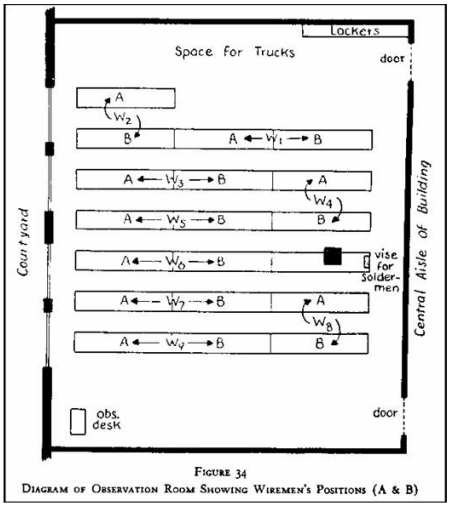


FIGURE 34
DIAGRAM OF OBSERVATION ROOM SHOWING WIREMEN'S POSITIONS (A & B)

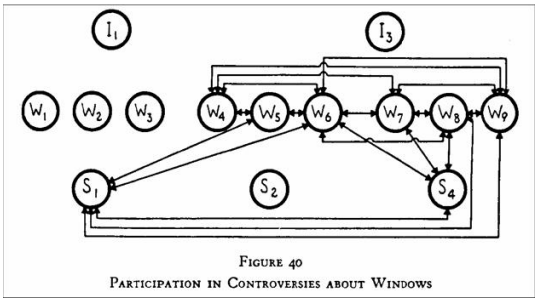


FIGURE 40
PARTICIPATION IN CONTROVERSIES ABOUT WINDOWS

Figure 3.1: Illustrations from an early social network study at the Hawthorne works of Western Electric in Chicago. The upper part shows the location of the workers in the wiring room, while the lower part is a network image of fights about the windows between workers (W), solderers (S) and inspectors (I).

important technique of network analysis, the sociogram is honored mostly for opening the way to a formal treatment of network analysis based on graph theory.

The following decades have seen a tremendous increase in the capabilities of network analysis mostly through new applications. SNA gains its relevance from applications and these settings in turn provide the theories to be tested and greatly influence the development of the methods and the interpretation of the outcomes. For example, one of the relatively new areas of network analysis is the analysis of networks in entrepreneurship, an active area of research that builds and contributes to organization and management science.

The vocabulary, models and methods of network analysis also expand continuously through applications that require to handle ever more complex data sets. An example of this process are the advances in dealing with longitudinal data. New probabilistic models are capable of modelling the evolution of social networks and answering questions regarding the dynamics of communities. Formalizing an increasing set of concepts in terms of networks also contributes to both developing and testing theories in more theoretical branches of sociology.

The increasing variety of applications and related advances in methodology can be best observed at the yearly Sunbelt Social Networks Conference series, which started in 1980.[4] The field of Social Network Analysis also has a journal of the same name since 1978, dedicated largely to methodological issues.[5] However, articles describing various applications of social network analysis can be found in almost any field where networks and relational data play an important role.

While the field of network analysis has been growing steadily from the beginning, there have been two developments in the last two decades that led to an explosion in network literature. First, advances in information technology brought a wealth of electronic data and significantly increased analytical power. We examine the possibilities of using electronic data for network analysis in Chapter 4. Second, the methods of SNA are increasingly applied to networks other than social networks such as the hyperlink structure on the Web or the electric grid. This advancement —brought forward primarily by physicists and other natural scientists— is based on the discovery that many networks in nature share a number of commonalities with social networks. In the following, we will also talk about networks in general, but it should be clear from the text that many of the measures in network analysis can only be strictly interpreted in the context of social networks or have very different interpretation in networks of other kinds.

3.3 Key concepts in network analysis

Social Network Analysis has developed a set of concepts and methods specific to the analysis of social networks. In the following, we introduce the most basic notions of network analysis and the methods we intend to use later in this book. We will proceed from

[4] See `http://www.insna.org/INSNA/sunbelt_inf.html`

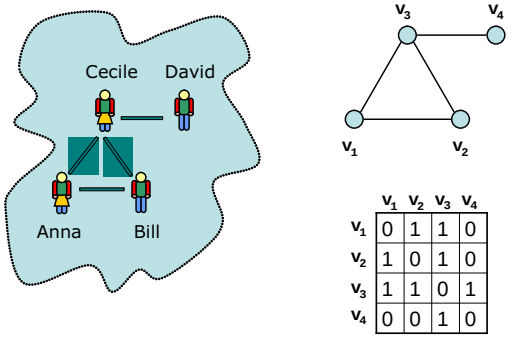
[5] See `http://www.elsevier.com/wps/find/journaldescription.cws_home/505596/description#description`

the global structure of networks toward the measurement of *ego-networks* (personal networks), i.e. from the macro level to the micro level of network analysis. For a complete reference to the field of social network analysis, we refer the reader to the exhaustive network analysis "Bible" of Wasserman and Faust [Wasserman et al., 1994]. Scott provides a shorter, but more accessible introductory text on network analysis [Scott, 2000].

3.3.1 The global structure of networks

As discussed above, a (social) network can be represented as a graph $G = (V, E)$ where V denotes the finite set of vertices and E denotes a finite set of edges such that $E \subseteq V \times V$. Recall that each graph can be associated with its characteristic matrix $M := (m_{i,j})_{n*n}$ where $n = |V|, m_{i,j} = \left\{ \begin{array}{l|l} 1 & (v_i, v_j) \in E \\ 0 & otherwise \end{array} \right\}$. Some network analysis methods are easier to understand when we conceptualize graphs as matrices (see Figure 3.2). Note that the matrix is symmetrical in case the edges are undirected. We will talk of a valued graph when we are also given a real valued weight function $w(e)$ defined on the set of edges, i.e. $w(e) := E \times \mathbb{R}$. In case of a valued graph, the matrix is naturally defined as $m_{i,j} = \left\{ \begin{array}{l|l} w(e) & (v_i, v_j) \in E \\ 0 & otherwise \end{array} \right\}$

Loops are not excluded in the above definition, although they rarely in occur in practical social network data sets. (In other words, the main diagonal of the matrix is usually empty.) Typically, we also assume that the network is connected, i.e. there is a single *(weak) component* in the graph.[6] Otherwise we choose only one of the components for analysis.



	v1	v2	v3	v4
v1	0	1	1	0
v2	1	0	1	0
v3	1	1	0	1
v4	0	0	1	0

Figure 3.2: Most network analysis methods work on an abstract, graph based representation of real world networks.

[6] A component is a maximal connected subgraph. Two vertices are in the same (strong) component if and only if there exists a (directed) path between them.

At this point the question of what else can we say about the structure of social networks arises naturally. Are there any commonalities to real world (social) networks that we could impose on our graph models or any kind of graph is equally likely to occur in practice? These questions are relevant because having a general model of social networks would allow us to answer questions in general, i.e. in a way that the answer should hold for all networks exhibiting the common characteristics. In some cases we could verify theories solely on an abstract model instead of having to collect network data. Further, understanding the global structure of networks can lead us to discover commonly occurring patterns of relationships or typical network positions worthy of formalizing in further detail.

An important clue about the structure of social networks came from a remarkable experiment by the American psychologist Stanley Milgram [Milgram, 1967]. Milgram went out to test the common observation that no matter where we live, the world around us seems to be *small*: we routinely encounter persons not known to us who turn out to be the friends of our friends. Milgram thus not only wanted to test whether we are in fact all connected but he was also interested in what is the average distance between any two individuals in the social network of the American society.

In order to find out he devised an experiment in which he gave letters to a few hundred randomly selected persons from Boston in Massachusetts and Omaha in the state of Nebraska. (The choice of these places was intentional: Milgram wanted to include a cross-section of the American society.) The participants were asked to send the letter to a single target person, namely a stockbroker in Sharon, Massachusetts. They were not allowed to send it directly to the target person, however. Instead every participant was asked to forward the letter to a person he or she knew on a first name basis. That person would then also need to follow the same instructions and send the letter to someone who was more likely to be acquainted with the stock broker. In the end, the letter would reach someone who knew the target person in question and would hand the letter to him. In other words, Milgram devised a chain-mail similar to the ones that now aimlessly circle the planet. However, these letters had a target and the chains stopped when they reached their final destination [Milgram, 1967].

Milgram calculated the average of the length of the chains and concluded that the experiment showed that on average Americans are no more than six steps apart from each other. While this is also the source of the expression *six degrees of separation* the actual number is rather dubious: not only was Milgram's sample too small, but even only 20% of the those letters have made it to their destination. Thus the number could be actually larger: those letters that did not make it would probably have resulted in longer paths. But the number could be also smaller as it is not guaranteed that the letters have travelled the shortest possible path from their source to the target. Still, Milgram's experiment had a tremendous impact on social network research and sociology as a whole as it showed that the number is orders of magnitude smaller than the size of the network.

Formally, what Milgram estimated is the size of the average shortest path of the network, which is also called *characteristic path length*. An open (simple) path in a graph is a sequence of vertices $v_{i_0}, v_{i_2}, \ldots, v_{i_n}$ such that $\forall j = 0 \ldots n-1 (v_{i_j}, v_{i_{j+1}}) \in E$ and $\forall j, k = v_{i_j} \neq v_{i_k}$, in other words every vertex is connected to the next vertex and no

vertex is repeated on the path.[7] The shortest path between two vertices v_s and v_t is a path that begins at the vertex v_s and ends in the vertex v_t and contains the least possible number of vertices. The shortest path between two vertices is also called a *geodesic*. The longest geodesic in the graph is called the diameter of the graph: this is the maximum number of steps that is required between any two nodes. The average shortest path is the average of the length of the geodesics between all pairs of vertices in the graph. (This value is not possible to calculate if the graph is not (strongly) connected, i.e. in case there exists a pair of vertices with no path between them.)

A practical impact of Milgram's finding is that we can exclude certain kind of structures as possible models for social networks. The two dimensional lattice model shown in Figure 3.3, for example, does not have the small world property: for a network of size n the characteristic path length is $2/3 * \sqrt{n}$, which is still too large a number to fit the empirical finding.

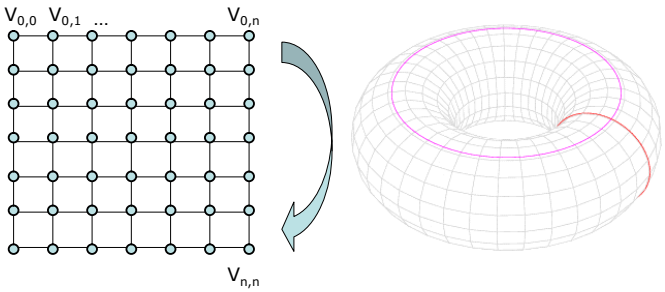


Figure 3.3: The 2D lattice model of networks (left). By connecting the nodes on the opposite borders of the lattice we get a toroidal lattice (right).

Another simple model could be the tree graph shown in Figure 3.4. However, a tree is unrealistic because it shows no *clustering*: we all know from practice that our friends are likely to know each other as well because we tend to socialize in groups. (If not for other reasons than other friends know each other because we introduced them to each other.) Clustering for a single vertex can be measured by the actual number of the edges between the neighbors of a vertex divided by the possible number of edges between the neighbors. When taken the average over all vertices we get to the measure known as *clustering coefficient*. The clustering coefficient of a tree is zero, which is easy to see if we consider that there are no triangles of edges (*triads*) in the graph. In a tree, it would never be the case that our friends are friends with each other.

The lattice and the tree also have the rather unappealing characteristic that every node has the same number of connections. We know from our everyday walks in life that some of us have much larger social circles than others. The *random graph* model proposed by the Hungarian mathematicians Erdős and Rényi offers an alternative. A random graph can be generated by taking a set of vertices with no edges connecting them. Subsequently,

[7] We assume that Milgram checked whether this is true for the paths that he found.

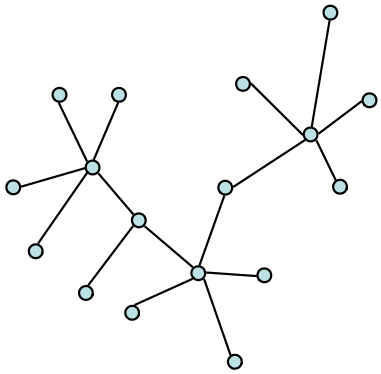


Figure 3.4: A tree is a connected graph where there are no loops and paths leading from a vertex to itself.

edges are added by picking pairs of nodes with equal probability. This way we create a graph where each pair of vertices will be connected with an equal probability. (This probability is a parameter of the process.)

If we continue the process long enough —we choose a high enough probability— the resulting random graphs will have a small characteristic path length and most likely exhibit some clustering. (Needless to say if we go on we end up with a complete graph.) Still, we can raise significant concerns against the cold probabilistic logic of a random graph. Due to limitations of space —if not for other reasons— we are unlikely to make friends completely in random from anywhere in the world. Although we meet strangers occasionally by sitting next to them on an airplane, we mostly socialize in a given geographic area and even then in limited social environments such as our work and living space. Again, the friends of our friends are likely to be our friends as well. If that happens in a random graph, it happens by accident.

Nevertheless, the Erdős-Rényi random graphs are interesting in the sense that they are examples of generative models. That is, random graphs are not (only) defined by what they are but also *how* they arise, i.e. the process of growing such a graph. These kinds of processes are also at the centerpoint of interest for the field of complex networks in physics where researchers study the emergence of complex global structures from systems that are defined solely through elementary interactions between primitive elements.

It is not so surprising then that the next steps in the search for a more appropriate network model came from physicists and mathematicians. In a seminal paper published in 1999 the mathematicians Steven Strogatz and Duncan Watts presented their alpha-model of networks [Watts, 1999]. In this model the network also grows, but not in a random fashion. In this model the number of mutual friends shared by any two nodes would determine the likelihood of a new tie forming. In the model of Watts and Strogatz a parameter alpha allows to fine-tune the exact influence of the number of friends on the

probability of a tie. The alpha model was successful in generating graphs with small path lengths and relative large clustering coefficients.

The alpha-model was later simplified by the authors in the so-called beta-model, which achieves the same results although in a somewhat less intuitive process [Watts and Strogatz, 1998]. The beta-model is also generative and it starts with a one-dimensional toroidal lattice where every node is connected not only to its neighbors but also to the neighbors of its neighbors. Consequently, a random edge is rewired by keeping one end of the edge fixed and reassigning the other end to another randomly selected node. We continue this process and rewire every link with a probability of beta, which is a parameter of the process. By choosing beta appropriately, the beta model allows to generate networks with small path lengths and relatively large clustering coefficients.

While the alpha and beta models could be considered perfect by these characteristics alone, they too fail to recreate an important feature of networks in nature: the scale-free characteristic of the degree distribution. The understanding of this phenomenon and the construction of a model that can reproduce it is due to another Hungarian, a physicist by the name of Albert-László Barabási [Albert-László Barabási and Réka Albert, 1999].

To understand the scale-free phenomenon we have to look at the degree-distribution of networks. Such a diagram shows how many nodes in the network have a certain number of neighbors (degrees). In a toroidal lattice all nodes have an equal number of neighbors. In the alpha and beta models as well as the random graphs of Erdős and Rényi this distribution is a normal distribution: there is an average degree, which is also the most common one. Degrees deviating from this are increasingly less likely. In real social networks, however, this distribution shows a different picture: the number of nodes with a certain degree is highest for small degree and the number of nodes with a given degree rapidly decreases for higher degrees. In other words, the higher the degree the least likely it is to occur. What is also surprising is the steepness of the distribution: the vast majority of the nodes have much fewer connections than the few *hubs* of the network. The exact correlation is a power law, i.e. $p(d) = d^{-k}$ where k is a parameter of the distribution.

As an example, the upper part of Figure 3.5 shows the degree distribution for the co-authorship network of the Semantic Web research community. The chart is a histogram showing how many people have a certain degree, i.e. a certain number of co-authors. The chart shows that most people have only one or two co-authors while only very few researchers have more than twenty co-authors. (One co-author is slightly less likely than two because the data comes from publications and the average publication has about two authors.) The lower part of the Figure shows the degree distribution of a random network of similar size.

Barabási not only discovered that this is a fundamental characteristic of many networks that he studied, but also gave a generative model to reproduce it. In this model, we start with a single node and add nodes in every step. The trick of Barabási is that when adding new nodes we link the node to an already existing node with a probability that is determined by how many edges the node already has. In other words, the rich get richer in this model: a node that has already attracted more edges than others will have a larger probability to attract even more connections in subsequent rounds. Barabási showed that

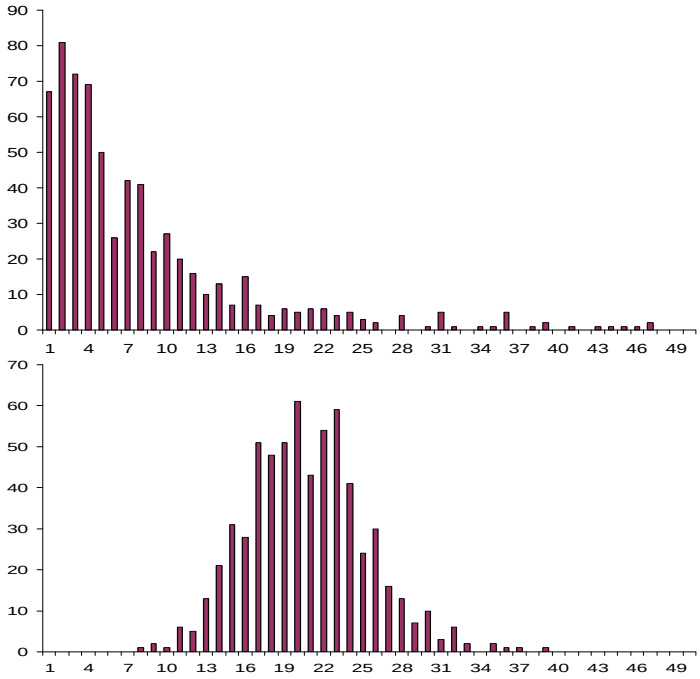


Figure 3.5: The degree distribution of a real world scale-free network (upper part) and the degree distribution of a random network of the same size (lower part).

the resulting networks can have a short characteristic path lengths and a large clustering coefficient as well as a degree distribution that approaches a power-law.

The works of Watts, Strogatz, Barabási and his colleagues are largely responsible for bringing network research into the scientific forefront. We note here that their work would have had a much more limited impact if its applications were only sociological. However, the small world phenomenon and scale-free characteristics are still routinely identified in many networks in nature such as transport networks within the cell. Even technological networks such as the hyperlink structure of the World Wide Web or the electrical transport network exhibit scale-free characteristics: the rich get richer logic creates scale-free structures in these networks by rewarding nodes proportionally to their existing number of links. By analyzing the model instead of particular instances of it allows scientists to formulate precise and general claims about these networks, for example with respect to their vulnerability to specific kind of attacks or the possibility for the spread of viruses or failures across them. In technology this impacted the design of networks and resulted, for example, in more efficient structures for peer-to-peer networks.

3.3.2 The macro-structure of social networks

Based on the findings about the global characteristics of social networks we now have a good impression about what they might look like. In particular, the image that emerges is one of dense clusters or social groups sparsely connected to each other by a few ties as shown in Figure 3.6. (These weak ties have a special significance as we will see in the following Section.) For example, this is the image that appears if we investigate the co-authorship networks of a scientific community. Bounded by limitations of space and resources, scientists mostly co-operate with colleagues from the same institute. Occasional exchanges and projects with researchers from abroad, however, create the kind of shortcut ties that Watts explicitly incorporated within his model. These shortcuts make it possible for scientists to reach each other in a relatively short number of steps.

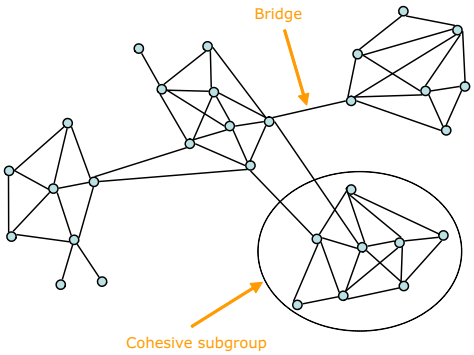


Figure 3.6: Most real world networks show a structure where densely connected subgroups are linked together by relatively few bridges

.

Particularly important are the hubs, the kind of individual such as Erdős was in the field of mathematics: he was one of the most prolific authors in the history of mathematics and co-authored with an unusual number of individuals. In fact, it is a common game in this field to calculate a researcher's Erdős-number, which is a measure of the number of steps in the co-authorship network from Erdős to the given researcher. Co-authors of Erdős are said to have an Erdős-number of one, their co-authors have a number of two etc.

Network visualizations based on topographic or physical principles can be helpful in understanding the group structure of social networks and pinpoint hubs that naturally tend to gravitate toward the center of the visualization. Unfortunately, computer generated network visualizations rarely show the kind of image seen in Figure 3.6. Displays based on multi-dimensional scaling, for example, attempt to optimize the visualization in a way that distances on the paper correlate with the distances between the nodes in the graph. However, with as few as four nodes it is easy to construct an example where there is no optimal solution to the placement problem. (Consider for example the 2D lattice shown

in Figure 3.3.) In general, the more dense the graph and the fewer the dimensions of the visualization the more likely the graph will degenerate into a meaningless "spaghetti bowl" tangle of nodes and edges.[8]

Fortunately, visualizations are also not the only kind of tools in the network analysts' toolkit to uncover subgroup structure. Various clustering algorithms exist for creating disjunct or overlapping subsets of nodes based on different definitions of a what a subgroup is or how to find one.

Several definitions are based on the observations that subgroups are densely connected and their members are close to each other in the graph. For example, a *clique* in a graph is maximal complete subgraph of three or more nodes. As complete subgraphs are very rare, the definition of a clique is typically relaxed by allowing some missing connections. For example, a *k-plex* is a maximal subgraph in which each node is adjacent to no fewer than $g_s - k$ nodes in the subgraph, where g_s is the number of nodes in the subgraph. The larger we set the parameter k, the larger the k-plexes that we will find. Other definitions constrain subgroups by putting limits on the maximum path length between the members.

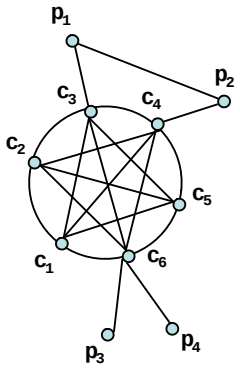
Yet another way of defining cohesiveness is to compare the density of ties within the group to the density of ties between members of the subgroup and the outside. The lambda-set analysis method we will use in our work is based on the definition of edge connectivity. Denoted with the symbol $\lambda(i, j)$, the edge connectivity of two vertices v_i and v_j is the minimum number of lines that need to be removed from a graph in order to leave no path between the two vertices. A lambda-set is then defined as a set of nodes where any pair of nodes from the set has a larger edge connectivity than any pair of nodes where one node is from within the set and the other node is from outside the set. Unlike the above mentioned k-plexes, lambda-sets also have the nice property that they are not overlapping.

The edge-betweenness clustering method of Mark Newman takes a different approach [Girvan and Newman,]. Instead of focusing on the density of subgroups, this algorithm targets the ties that connect them. The ties that are in between groups can be spotted by calculating their *betweenness*. The betweenness of an edge is calculated by taking the set of all shortest paths in the graph and looking at what fraction of them contains the given edge. An edge between clusters has a much higher betweenness than edges inside clusters because all shortest paths between nodes in the different clusters have to go through the given edge. By progressively removing the edges with the highest betweenness the graph falls apart in distinct clusters of nodes.

Clustering a graph into subgroups allows us to visualize the connectivity at a group level. In some cases we already have an idea of what this macro-structure might look like. A typical pattern that often emerges in social studies is that of a *Core-Periphery (C/P) structure*. A C/P structure is one where nodes can be divided in two distinct subgroups: nodes in the core are densely connected with each other and the nodes on the periphery, while peripheral nodes are not connected with each other, only nodes in the core (see Figure 3.7). The matrix form of a core periphery structure is a $\begin{pmatrix} 1 & . \\ . & 0 \end{pmatrix}$ matrix. Al-

[8] In fact, when used inappropriately, visualizations are among the most dangerous tools of a social network analyst: at worst they not only obscure important phenomena but also allow to argue false theories.

gorithms for identifying C/P structures and other *block models* (structural patterns) work by dividing the set of nodes in a way that the error the between the actual image and the "perfect" image is minimal. The result of the optimization is a classification of the nodes as core or periphery and a measure of the error of the solution.



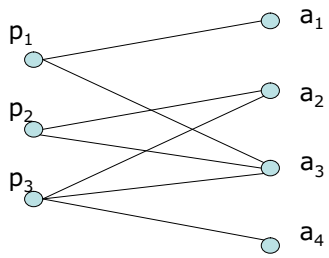
c1	c2	c3	c4	c5	c6	p1	p2	p3	p4
1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	0	0	0
1	1	1	1	1	1	0	1	0	0
1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	0	0	1	1
0	0	1	0	0	0	0	1	0	0
0	0	0	1	0	0	1	0	0	0
0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	0	0

Figure 3.7: A Core-Periphery structure that would be perfect without the edge between nodes p_1 and p_2.

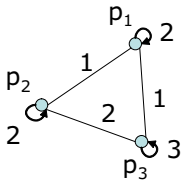
We will also encounter situations in our work where it is some additional information that allows us to group our nodes into categories. For example, in the case of a scientific community we might have data on the interests or affiliations of researchers. When we have such attribute data available we already have a division of our network into clusters based on shared interests or affiliations. These clusters are overlapping depending on whether a single person is allowed to have multiple interests or affiliations.

As it is fairly common to have attribute information on subjects besides the relational data, the study of *affiliation networks* is an important topic in network analysis. Affiliation networks contain information about the relationships between two sets of nodes: a set of subjects and a set of affiliations. An affiliation network can be formally represented as a *bipartite graph*, also known as a *two-mode network*. In general, an n-partite graph or n-mode network is a graph $G = \langle V, E \rangle$ where there exists a partitioning $V = \bigcup_{i=1}^{n} V_i$ such that $\bigcap_{i=1}^{n} V_i = 0$ and $(V_i \times V_i) \cap E = 0$. In other words, the set of vertices is divided into n disjoint sets and there are no edges between vertices belonging to the same set.

There are relative few methods of analysis that are specific to affiliation networks; when dealing with affiliation networks they are typically transformed directly to a regular, one-mode network. This transformation considers the overlaps between the affiliations as a measure of tie strength between the actors (see Figure 3.8). For example, we can generate a network among scientists by looking at how many interests they have in common. We would place an edge between two researchers if they have interests in common and would weight the edge according to the number of shared interests. The analysis of such a one-mode network would follow as usual.



	p_1	p_2	p_3	a_1	a_2	a_3	a_4
p_1	0	0	0	1	0	1	0
p_2	0	0	0	0	1	1	0
p_3	0	0	0	0	1	1	1
a_1	1	0	0	0	0	0	0
a_2	0	1	1	0	0	0	0
a_3	1	1	1	0	0	0	0
a_4	0	0	1	0	0	0	0



	p_1	p_2	p_3
p_1	2	1	1
p_2	1	2	2
p_3	1	2	3

Figure 3.8: A two-mode network of persons and affiliation (shown as a graph and a matrix above) can be folded into a regular social network by considering the overlaps in affiliations (graph and matrix below).

Interestingly, we can also use this technique to map the relationship between the affiliations themselves. In this case, we create a network of the affiliations as the nodes and draw edges between the nodes based on the number of individuals who share that affiliations. This technique is commonly used, for example, to map *interlocking directorates*, overlaps in the board membership of companies. In such a setting the analysis starts with data on persons and their positions on the boards of various companies. Then a network of companies is drawn based on potential interactions through shared members on their boards. The website TheyRule[9] shows the potential conflicts of interests in the American business, political and media elite by charting the connectivity of individuals and institutions using this technique (see Figure 3.9).

3.3.3 Personal networks

In many cultures the term *networking* gained a negative connotation as a reference to nepotism, the unfair access to advantages through "friends of friends". However, among others in the American society being a good "networker" has become to be seen as an important personal skill that every professional should master.

These terms reflect one of the basic intuitions behind network analysis: that social structure plays a key role in defining our opportunities and constraints with respect to access to key resources. In most settings network ties serve as important conduits for

[9] `http://www.theyrule.net`

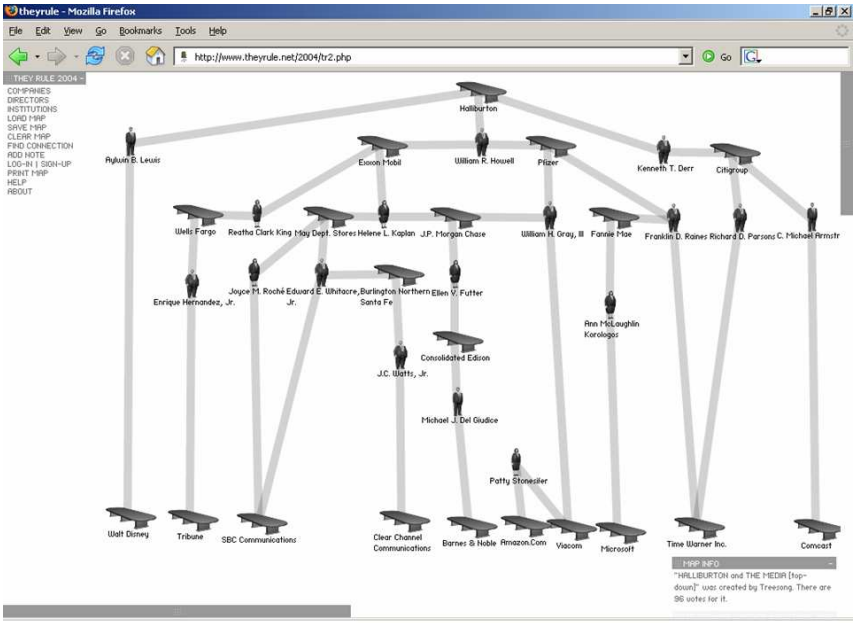


Figure 3.9: The site *theyrule.net* visualizes the interconnectedness of the American political, business and media elite by showing connections between executive boards of various institutions and their board members. Connections between individuals can be inferred by the boards in which they co-participate.

knowledge and other intangible resources such as social support. More often than not establishing social links to those who control tangible resources in a community provides competitive advantages to the individual, especially if those resources are scarce. In turn, a person with such access to key players will be increasingly sought after by other individuals. Conversely, the absence of such relations means the lack of access to the intangible resources "flowing" through social networks and to those tangible resources controlled by certain individuals or groups.

In the following we summarize different dimensions of *social capital* and some related concepts and measures based on the framework of Nahapiet and Ghoshal [Nahapiet and Ghoshal, 1998]. We choose this well-known framework as it describes social capital in relation to its function in the development of knowledge and knowing capability[10], which they call *intellectual capital*. In particular, they suggest that various aspects of social capital enable the creation of intellectual capital by allowing exchanges to take place that lead to the combination of knowledge (either incrementally or radically) possessed by individuals.

[10]The distinction references the long standing debate on the two forms of knowledge: objective, analytical and explicit knowledge versus subjective, experience-based and tacit.

We hypothesize that social capital as a factor in building intellectual capital is key to the performance of individuals in research communities such as the Semantic Web that we investigate in Chapter 9. Note also that not only social capital can lead to intellectual capital but also the other way around, which is clearly visible in scientific communities. Consider that the very existence of a Semantic Web community is due to the accumulation of knowledge in the fields of Knowledge Representation and Web science and the understanding of the potential combination of the two. In effect, there is a complex feedback cycle where the intellectual capital brings researchers together leading to further socialization, which in turn leads to the creation of new intellectual capital.

Note that in the following we focus on the individual level of analysis. Although Nahapiet and Ghoshal extend intellectual capital to the collective level they note that there is significant discussion in the literature as to whether collectives can be attributed intellectual capital beyond the knowledge of their members and if yes, how the two levels of analysis are related.

Nahapiet and Goshal define social capital as the sum of actual and potential resources embedded within, available through, and derived from the network of relationships possessed by an individual or social unit [Nahapiet and Ghoshal, 1998]. They identify three main dimensions: the structural, relational and cognitive dimensions of social capital. Common to these dimensions is that they constitute some aspect of the social structure and that they facilitate actions of individuals.

The **structural dimension** of social capital refers to patterns of relationships or positions that provide benefits in terms of accessing large, important parts of the network. Common to structural measures of social capital is that they put a single node (the ego) in the center and provide a measure for the node based on his connectivity to other nodes (the alters).

A simple, but effective measure is the *degree centrality* of the node. Degree centrality equals the graph theoretic measure of degree, i.e. the number of (incoming, outgoing or all) links of a node. This measure is based on the idea that an actor with a large number of links has wider and more efficient access to the network, less reliant on single partners and because of his many ties often participates in deals as a third-party or broker. Degree centrality does not take into account the wider context of the ego, and nodes with a high degree may in fact be disconnected from large parts of the network. However, the degree measure features prominently in the scale-free model, which makes it an important measure to investigate.

A second, more intuitive measure of centrality is *closeness centrality*, which is obtained by calculating the average (geodesic) distance of a node to all other nodes in the network. Closeness centrality thus characterizes the reach of the ego to all other nodes of the network. In larger networks it makes sense to constrain the size of the neighborhood in which to measure closeness centrality. It makes little sense, for example, to talk about the most central node on the level of a society. The resulting measure is called *local closeness centrality*.

Two other measures of power and influence through networks are related to the similar advantages of *broker positions* and *weak ties*. A broker gains advantage by standing in between disconnected communities. As we have seen above ties spanning communities tend to be sparse compared to the dense networks of cohesive subgroups and as a result

there are typically few *bridges* across communities. Brokers controlling these bridges are said to be in an advantageous position especially because of the value attributed to the information flowing across such ties.

The measure of *betweenness centrality* identifies broker positions by looking at the extent to which other parties have to go through a given actor to conduct their dealings. Consequently, betweenness is defined as the proportion of paths —among the geodesics between all pairs of nodes— that pass through a given actor. (Betweenness centrality is measured in a similar way as edge-betweenness, but it is a measure of a node, not of an edge.) As with closeness centrality it is often desirable to compute betweenness in a fixed neighborhood of the ego.

The more complex measure of Ronald Burt is related to the idea of *structural holes* [Burt, 1995]. A structural hole occurs in the space that exists between closely clustered communities. According to Burt, a broker gains advantage by bridging such holes. Therefore this measure favors those nodes that connect a significant number of powerful, sparse linked actors with only minimal investment in tie strength. Burt suggests that structural holes show information benefits in three forms: access to large, disparate parts of the network (knowing where to turn for information, who can use some information), timing (receiving information sooner than others) and reputation through referrals.

In contrast to Burt, Coleman stresses the importance of a dense network with strong ties [Coleman, 1988]. In such a network knowledge can be easily shared because of the high level of general trust and the presence of well-established norms. Density can be measured by computing the clustering coefficient measure introduced above on a neighborhood of the individual.

Researchers intending to reconcile these views emphasize that Burt and Coleman type network often serve different, complementary purposes in practice. Further, there is evidence that these are extreme views: overly dense networks can lead to *overembeddedness* [Gargiulo and Benassi, 2000]. On the other hand, structural embeddedness is necessarily present as through transitivity our friends are likely to develop ties of their own over time.

The aspect of tie strength is an example of the **relational dimension** of social capital, which concerns the kind of personal relationships that people have developed with each other through a history of interaction [Granovetter, 1992]. The relationships of pairs of actors who occupy similar network positions in similar network configurations may significantly differ based on their past interactions and thus their possibilities for action might also differ.

The benefits of *weak ties* have been exposed by a renowned study in network analysis carried out by Mark Granovetter and described in his paper, *The Strength of Weak Ties* [Granovetter, 1973]. In this first study, Granovetter looked at two Boston communities and their responses to the threat of urban development. He came to the conclusion that efficient response was not so much a function of a dense, strong tie network but it rather depended on occasional “weak” ties between individuals who only saw each other occasionally. In a second study he found the same effect when looking at how different individuals mobilize their social networks when finding a job. While it has been known long before that most jobs are found through personal connections, Granovetter showed

that somewhat surprisingly close friends play a minor role in this process compared to casual acquaintances.

The discussion on the advantages of weak ties versus strong ties parallels the discussion on Burt and Coleman type networks: the connection between the two dimensions is that the bridge ties that we build to other social groups than our own tend to be weak ties as we invest much less of our energy in building and maintaining them. In particular, from the limited time available for social interaction we spend most in building strong ties to our immediate social circles of close colleagues, friends and family. It is a well-known adage in social science that social interaction creates similarity and similarity creates social interaction: we seek out those who are similar to us and in the process of interaction our peers become (even more) similar to us. As a result, our immediate contacts are likely to be "birds of a feather": individuals with similar resources, knowledge, ideas and social access and thus unlikely the provide the informational advantage required for successful job search. On the other hand, the personal trust, the accumulated experiences, the mutual lasting obligations that characterize strong ties reduce the cost (by reducing the risk) of many exchange transactions [Putnam, 1993].

Less attention is devoted in the literature of social capital to the last, **cognitive dimension** of the Nahapiet-Goshal framework. The cognitive dimension refers to those resources providing shared representations, interpretations and systems of meaning [Cicourel, 1973]. In particular, cognitive ties are based on the existence of shared languages, signs and narratives, which facilitate the exchange of knowledge. However, excessive cognitive similarity (associated with strong ties) is likely to lead to cognitive overembeddedness.

We investigate the distinct advantages of certain forms of cognitive embeddedness in our study of the Semantic Web community. In brief, we hypothesize that the access to a variety of cognitive contexts positively contributes to the performance of researchers beyond the well-known advantages of certain forms of structural embeddedness, i.e. that in fact the cognitive dimension is a separate and relevant dimension for assessing the ability of building intellectual capital. We will measure the impact of the cognitive dimension of social capital by comparing the performance of researchers with cognitively heterogeneous personal networks and cognitively homogeneous personal networks (see Chapter 9).

The reader might wonder at this point how these measures of various aspects of social capital are used in analysis. We have discussed in the beginning that the approach of network analysis is statistical and making claims about individual cases does not fit that approach. In fact, these measures are typically used in a "cases-by-variables" analysis where these network measures are correlated with some output variable using regression analysis. Besides network variables other attributes of the individuals are also entered into the model as control variables (see Figure 3.10).

For example, a typical study of this kind is the one carried out by Burt to verify that his structural holes measure can predict the creativity of supply chain managers at a large company. Burt's idea is that individual who have contacts spanning structural holes have better ideas due to their extended vision of the organization [Burt, 2004]. Burt proves this idea by first mapping the organizational network and calculating the structural holes measure for every individual involved in the study. Subsequently he performs a study

case	var_1	var_2	...	var_n	var_o
Anna	0.92	0.23	...	0.37	3.2
Bill	1.73	...			...
Cecile	...				
David					

$$\lambda_1 var_1 + \cdots + \lambda_n var_n + c = var_o + \varepsilon$$

Figure 3.10: In a cases-by-variables analysis we fit a linear equation (below) to the model described by the table containing the values of the variables for each case.

to measure the creativity of the supply chain managers by asking them to generate ideas about improving their work and then asking other, independent parties to rate those ideas. Lastly, he proves that the structural holes measure correlates with creativity by establishing a linear equation between the network measure and the individual characteristics on one side of the equation and creativity on the other side.

3.4 Discussion

In this Chapter we have introduced the development and key concepts of Social Network Analysis, which we will apply later in our case studies (Chapter 8,9 and 10. Social Network Analysis is a branch of sociology that is formalized to a great extent based on the mathematical notions of graph theory (and to a lesser degree matrix algebra and probability theory). This formal model captures the key observation of Social Network Analysis, namely that to a great extent social structure alone determines the opportunities and limitations of social units and ultimately effects the development of the network as well as substantial outputs of the community.

Social Network Analysis already enjoys many of the benefits of increasing formalization. Formerly vague theoretical concepts have been given a crisp, verifiable definition based on the network model of communities. This formality served network analysis to reduce the ambiguity in formulating and testing its theories and contributed to more coherence in the field by allowing researchers to reliably build on previous results.

In this book we will argue for further formalization in Network Analysis in order to capture more of the semantics of network data. We will argue that this is a necessary step to exploit the increasing availability of network data in electronic formats and to counter the need to combine such sources of information for analysis. In the next Chapter we will show some examples of the emerging possibilities in collecting social network data from the Internet. We will discuss the possibilities of semantic-based representations of electronic data in Chapters 5 and 6.

Part II

Semantic Technology for Social Network Analysis

Chapter 4

Electronic sources for network analysis

From the very beginning of the discipline collecting data on social networks required a certain kind of ingenuity from the researcher. First, social networks have been studied by observation. The disadvantage of this method is the close involvement of the researcher in the process of data collection. Standardized surveys minimize (but do not completely eradicate) the influence of the observer but they rely on an active engagement of the population to be studied. Unfortunately, as all of us are flooded these days by surveys of all kinds, achieving a high enough response rate for any survey becomes more and more problematic. In some settings such as within companies surveys can be forced on the participants, but this casts serious doubts on whether the responses will be spontaneous and genuine. Worse yet, observations and surveys need to be repeated multiple times if one would like to study network dynamics in any detail.

Data collection using these manual methods are extremely labor intensive and can take up to fifty per cent of the time and resources of a project in network analysis. Oftentimes the effort involved in data collection is so immense that network researchers are forced to reanalyze the same data sets over and over in order to be able to contribute to their field.

Network analysts looking for less costly empirical data are often forced to look for alternatives. A creative solution to the problem of data collection is to reuse existing electronic records of social interaction that were not created for the purposes of network analysis on the first place. Scientific communities, for example, have been studied by relying on publication or project databases showing collaborations among authors or institutes [Barabási et al., 2002, Grobelnik and Mladenic, 2002]. Official databases on corporate technology agreements allow us to study networks of innovation [Lemmens, 2003], while newspaper archives are a source of analysis for studies on topics ranging from the role of social-cognitive networks in politics [van Atteveldt et al., 2006] to the structure of terror organizations [Krebs, 2002]. These sources often support dynamic studies through historical analysis. Nevertheless, the convenience comes at a price: access to publication

and patent databases, media archives, legal and financial records often carries a significant price tag.

However, there is one data source that is not only vast, diverse and dynamic but also free for all: the Internet. In the following, we look at a sample of works from the rapidly emerging field of *e-social science*. Common to these studies is that they rely entirely on data collected from electronic networks and online information sources, which allows a complete automation of the data collection process. None of these works rely on commercial databases and yet many of them are orders of magnitude larger than studies based on data collected through observation or surveys. They represent a diversity of social settings and a number of them also exploit the dynamics of electronic data to perform longitudinal analysis. We will spend more attention on methods of social network extraction from the Web that we use in our analysis of the Semantic Web community (Chapter 9).

There are limits of course to the potential of e-social science. Most trivially, what is not on the Web can not be extracted from the Web, which means that there are a number of social settings that can only be studied using offline methods. There also technological limits to the accuracy of any method that relies on Information Extraction. For these reasons it is natural to evaluate our methods before using them for network analysis. We return to this issue in Chapter 8.

4.1 Electronic discussion networks

One of the foremost studies to illustrate the versatility of electronic data is a series of works from the Information Dynamics Labs of Hewlett-Packard.

Tyler, Wilkinson and Huberman analyze communication among employees of their own lab by using the corporate email archive [Tyler et al., 2003]. They recreate the actual discussion networks in the organization by drawing a tie between two individuals if they had exchanged at least a minimum number of total emails in a given period, filtering out one-way relationships. Tyler et al. find the study of the email network useful in identifying leadership roles within the organization and finding formal as well as informal communities. (Formal communities are the ones dictated by the organizational structure of the organization, while informal communities are those that develop across organizational boundaries.) The authors verify this finding through a set of interviews where they feed back the results to the employees of the Lab.

Wu, Huberman, Adamic and Tyler use this data set to verify a formal model of information flow in social networks based on epidemic models [Wu et al., 2004]. In yet another study, Adamic and Adar revisits one of the oldest problems of network research, namely the question of *local search*: how do people find short paths in social networks based on only local information about their immediate contacts? Their findings support earlier results that additional knowledge on contacts such as their physical location and position in the organization allows employees to conduct their search much more efficiently than using the simple strategy of always passing the message to the most connected neighbor.

Despite the versatility of such data, the studies of electronic communication networks based on email data are limited by privacy concerns. For example, in the HP case the content of messages had to be ignored by the researchers and the data set could not be shared with the community. In the work of Peter Gloor and colleagues, the source of these data for analysis is the archive of the mailing lists of a standard setting organization, the World Wide Web Consortium (W3C) [Gloor et al., 2003]. The W3C —which is also the organization responsible for the standardization of Semantic Web technologies— is unique among standardization bodies in its commitment to transparency toward the general public of the Internet and part of this commitment is the openness of the discussions within the working groups. (These discussion are largely in email and to a smaller part on the phone and in face-to-face meetings.)

Group communication and collective decision taking in various settings are traditionally studied using much more limited written information such as transcripts and records of attendance and voting, see e.g. As in the case with emails Gloor uses the headers of messages to automatically re-create the discussion networks of the working group.[1] The main technical contribution of Gloor is a dynamic visualization of the discussion network that allows to quickly identify the moments when key discussions take place that activate the entire group and not just a few select members. Gloor also performs a comparative study across the various groups based on the structures that emerge over time.

Although it has not been part of this work, it would be even possible to extend such studies with an analysis of the role of networks in the decision making process as voting records that are also available in electronic formats. Further, by applying emotion mining techniques from AI to the contents of the email messages one could recover agreements and disagreements among committee members. Marking up the data set manually with this kind of information is almost impossible: a single working group produces over ten thousand emails during the course of its work.

4.2 Blogs and online communities

Content analysis has also been the most commonly used tool in the computer-aided analysis of blogs (web logs), primarily with the intention of trend analysis for the purposes of marketing.[2] While blogs are often considered as “personal publishing” or a “digital diary”, bloggers themselves know that blogs are much more than that: modern blogging tools allow to easily comment and react to the comments of other bloggers, resulting in webs of communication among bloggers. These discussion networks also lead to the establishment of dynamic communities, which often manifest themselves through syndicated blogs (aggregated blogs that collect posts from a set of authors blogging on similar topics), blog rolls (lists of discussion partners on a personal blog) and even result in real

[1] A slight difference is that unlike with personal emails messages to a mailing list are read by everyone on the list. Nevertheless individuals interactions can be partly recovered by looking at To: and CC: fields of email headers as well as the Reply-To field.

[2] See for example the works presented at the 2006 AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs at `http://www.umbriacom.com/aaai2006_weblog_symposium/` or the Workshop on the Weblogging Ecosystem at `http://wwe2005.blogspot.com/`

world meetings such as the Blog Walk series of meetings[3]. Figure 4.1 shows some of the features of blogs that have been used in various studies to establish the networks of bloggers.

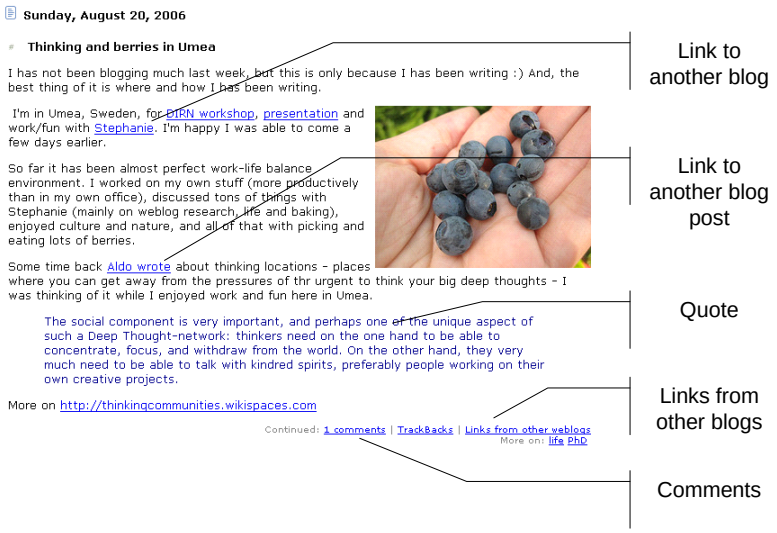


Figure 4.1: Features of blogs that can be used for social network extraction. Note also that — unlike web pages in general— blog entries are timestamped, which allows to study network dynamics, e.g. the spread of information in online communities.

Blogs make a particularly appealing research target due to the availability of structured electronic data in the form of RSS (Rich Site Summary) feeds. RSS feeds contain the text of the blog posts as well as valuable metadata such as the timestamp of posts, which is the basis of dynamic analysis. For example, Kumar et al. and Gruhl et al. study information diffusion in blogs based on this information [Kumar et al., 2003, Gruhl et al., 2004]. The early work of Efimova and Anjewierden also stands out in that they were among the first to study blogs from a communication perspective [Anjewierden and Efimova, 2006]. Adar and Adamic offer a visualization of such communication in blogs in [Adar and Adamic, 2005].

The 2004 US election campaign represented a turning point in blog research as it has been the first major electoral contest where blogs have been exploited as a method of building networks among individual activists and supporters (see for example [Adamic and Glance, 2005]). Blog analysis has suddenly shed its image as relevant only to marketers interested in understanding product choices of young demographics; following this campaign there has been explosion in research on the capacity of web logs for creating and maintaining stable, long distance social networks of different kinds.

[3] `http://blogwalk.interdependent.biz/wikka.php?wakka=HomePage`

Since 2004, blog networks have been the object of study for a number of papers in the blog research track of the yearly Sunbelt social networks conference.

Online community spaces and social networking services such as MySpace, LiveJournal cater to socialization even more directly than blogs with features such as social networking (maintaining lists of friends, joining groups), messaging and photo sharing.[4] As they are typically used by a much younger demographic they offer an excellent opportunity for studying changes in youth culture. Paolillo, Mercure and Wright offer a characterization of the LiveJournal community based on the electronic data that the website exposes about the interests and social networks of its users [Paolillo et al., 2005]. Their study is an excellent example of how directly available electronic data enables the longitudinal analysis of large communities (more than 10,000 users).

LiveJournal exposes data for research purposes in a semantic format, but unfortunately this is the exception rather than the norm. Most online social networking services (Friendster, Orkut, LinkedIn and their sakes) closely guard their data even from their own users. (Unless otherwise stated these data provided to an online service belongs to the user. However, most of these services impose terms of use that limit the rights of their users.) A technological alternative to these centralized services is the FOAF network (see also Chapter 6). FOAF profiles are stored on the web site of the users and linked together using hyperlinks. The drawback of FOAF is that at the moment there is a lack of tools for creating and maintaining profiles as well as useful services for exploiting this network. Nevertheless, a few preliminary studies have already established that the FOAF network exhibits similar characteristics to other online social networks [Paolillo and Wright, 2004, Ding et al., 2005].

4.3 Web-based networks

The content of Web pages is the most inexhaustible source of information for social network analysis. This content is not only vast, diverse and free to access but also in many cases more up to date than any specialized database. On the downside, the quality of information varies significantly and reusing it for network analysis poses significant technical challenges. Further, while web content is freely accessible in principle, in practice web mining requires efficient search that at the moment only commercial search engines provide.[5]

There are two features of web pages that are considered as the basis of extracting social relations: links and co-occurrences (see Figure 4.2). The linking structure of the Web is considered as proxy for real world relationships as links are chosen by the author of the page and connect to other information sources that are considered authoritative and relevant enough to be mentioned. The biggest drawback of this approach is that such direct links between personal pages are very sparse: due to the increasing size of the Web

[4] In July, 2006 MySpace has passed Google and Yahoo! as the most popular website on the Web, see `http://www.techcrunch.com/2006/07/11/myspace-hit-1-us-destination-last-week-hitwise/`

[5] Unfortunately, search engines such as Google or Yahoo! typically limit the number of queries that can be issued a day. There has been work on an open source search engine since 2003, see `http://lucene.apache.org/nutch/`. However, to this day this effort did not result in an alternative to commercial search engines

searching has taken over browsing as the primary mode of navigation on the Web. As a result, most individuals put little effort in creating new links and updating link targets or have given up linking to other personal pages altogether.

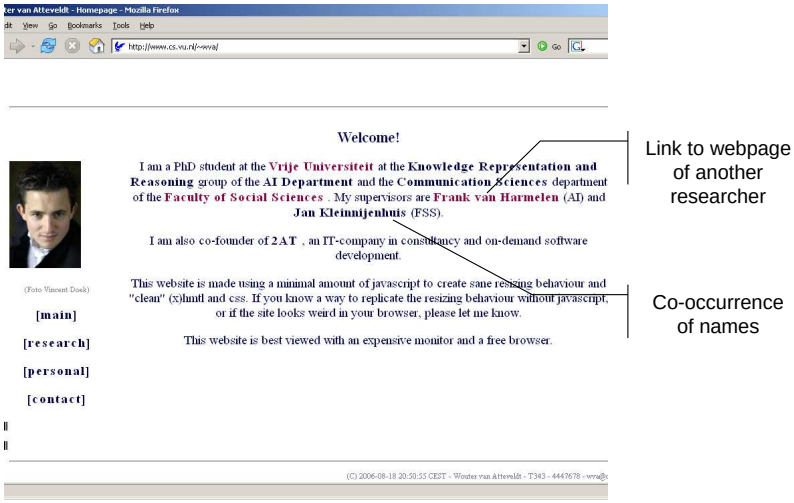


Figure 4.2: Features in web pages that can be used for social network extraction.

For this reason most social studies based on the linking structure of the web are looking at relationships at higher levels of aggregation. For example, members of the EICSTES project investigate the web connectivity of entire scientific institutions for the purposes of *webometrics* or web-based scientometrics in the EICSTES project[6]. For example, Heimeriks, Hörlesberger and Van den Besselaar compare communication and collaboration networks across different fields of research using a multi-layered approach [Heimeriks et al., 2003]. The data for this analysis comes from bibliographic records, project databases and hyperlink networks. The connections for the latter are collected by crawling the websites of the institutions involved. In principle it could be possible to extract more fine-grained networks from the homepages of the individual researchers. However, links between homepages are too sparse to be analyzed on their own and automating this task would also require solving what is known as the home page search problem: locating the homepage of individuals given their name and description.

Co-occurrences of names in web pages can also be taken as evidence of relationships and are a more frequent phenomenon. On the other hand, extracting relationships based on co-occurrence of the names of individuals or institutions requires web mining as names are typically embedded in the natural text of web pages. (Web mining is the application of text mining to the content of web pages.) The techniques employed here are statistical methods possibly combined with an analysis of the contents of web pages.

[6] `http://www.eicstes.org/`

Web mining has been first tested for social network extraction from the Web in the work of Kautz el al. on the ReferralWeb project in the mid-90s [Kautz et al., 1997]. The goal of Kautz et al. was not to perform sociological experiments but to build a tool for automating what he calls *referral chaining*: looking for experts with a given expertise who are close to the user of the system, i.e. experts who can be accessed through a chain of referrals. An example of a question that could be asked to the system is "show me all experts on simulated annealing who are at most three steps away from me in the network."

As the authors were researchers themselves, they were primarily interested in solving the referral chaining problem in the scientific domain where finding experts on a given topic is a common problem in organizing peer-reviews. (Kautz also applied his system later toward recommending experts in a corporate setting at AT&T.) The ReferralWeb system was bootstrapped with the names of famous AI researchers. The system extracted connections between them through co-occurrence analysis. Using the search engine Altavista the system collected page counts for the individual names as well as the number of pages where the names co-occurred. Note that this corresponds to a very shallow parsing of the web page as indirect references are not counted this way (e.g. the term "the president of the United States" will not be associated with George Bush even if he was mentioned as the president elsewhere in the text.)

Tie strength was calculated by dividing the number of co-occurrences with the number of pages returned for the two names individually (see Figure 4.3). Also known as the Jaccard-coefficient, this is basically the ratio of the sizes of two sets: the intersection of the sets of pages and their union [Salton, 1989]. The resulting value of tie strength is a number between zero (no co-occurrences) and one (no separate mentioning, only co-occurrences). If this number has exceeded a certain fixed threshold it was taken as evidence for the existence of a tie.

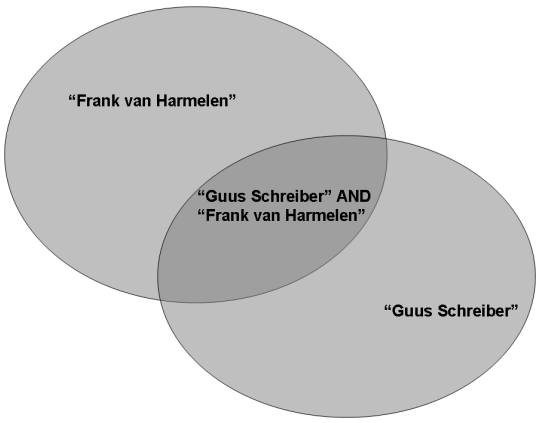


Figure 4.3: The Jaccard-coefficient is the ratio of the intersection and the union of two sets. In the case of co-occurrence analysis the two sets contain the pages where the individual names occur. The intersection is formed by the pages where both names appear.

Although Kautz makes no mention of it we can assume that he also filtered ties also based on support, i.e. the number of pages that can be found for the given individuals or combination of individuals. The reason is that the Jaccard-coefficient is a relative measure of co-occurrence and it does not take into account the absolute sizes of the sets. In case the absolute sizes are very low we can easily get spurious results: for example, if two names only occur once on the Web and they occur on the same page, their coefficient will be one. However, in this case the absolute sizes are too low to take this as an evidence for a tie.

The expertise of individuals was extracted by looking for capitalized phrases that appeared in documents returned by the search engine that were not proper names. The network in the system has grown two ways. Firstly, the documents from the Web were searched for new names using *proper name extraction*, a fairly reliable NLP technique. These names were then used to extract new names, a process that was repeated two or three times. (Note that this is similar to the *snowballing technique* of network analysis where the network under investigation is growing through new names generated by participants in the study.) Second, users of the system were also allowed to register themselves.

Kautz never evaluated his system in the sense of asking whether the networks he extracted are an accurate reflection of real world networks. He notes that the system as a recommender system performed well on both the research domain and in the corporate setting, although "the recommendations made by (any) recommender system tend to be either astonishingly accurate, or absolutely ridiculous [, which is] true for any AI-complete problem". However, he suggest that the system is able to keep the trust of the user provided that it is made transparent. For example, the system can show the evidence on which the recommendation is based and indicate the level of confidence in its decisions. With respect to the corporate setting Kautz also notes that the results in principle can be better than using the official corporate records for locating experts as personal pages are often more up-to-date. In the scientific setting such records are non-existent and even if there existed a central system where experts can describe their social networks and expertise it would be just as likely to become obsolete on the long term as corporate yellow pages are.

In our work we use the basic method of Kautz in a slightly different way. Since our goal is the extraction of social networks we are given a list of names to begin with. We consult the search engine for investigating the possible tie between all pairs of names. Note that the number of queries required grows quadratically with the number of names, which is not only costly in terms of time but is limited by the number of queries that search engines allow. While this is not a problem in our case study, optimizations are required for larger scale analysis. A solution is proposed by Matsuo et al. who recreate the original method of Kautz by first extracting possible contacts from the results returned by the search engine for the individual names [Matsuo et al., 2006]. This significantly reduces the number of queries that need to be made to the search engine at a minimal loss.

We also experiment with different measures of co-occurrence. A disadvantage of the Jaccard-coefficient is that it penalizes ties between an individual whose name often occurs on the Web and less popular individuals (see Figure 4.4). In the science domain

this makes it hard to detect, for example, the ties between famous professors and their PhD students. In this case while the name of the professor is likely to occur on a large percentage of the pages of where the name of the PhD student occurs but not vice versa. For this reason we use an asymmetric variant of the coefficient. In particular, we divide the number of pages for the individual with the number of pages for both names and take it as evidence of a directed tie if this number reaches a certain threshold. We experiment with choosing an appropriate value for this threshold and the threshold for tie strength in Chapter 8.

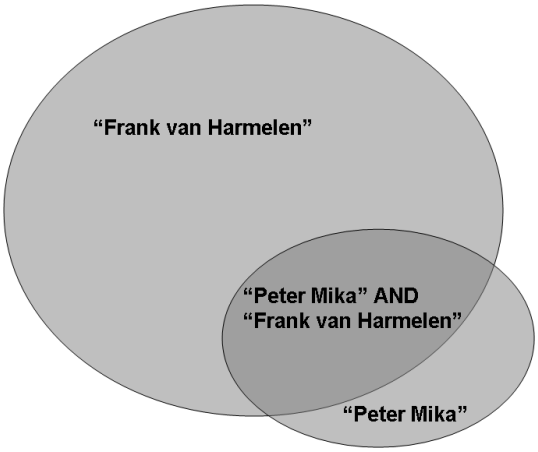


Figure 4.4: The Jaccard-coefficient does not show a correlation in cases where there is a significant difference in the sizes of the two sets such as in the case of a student and a professor.

Second, we associate researchers with topics in a slightly different way. In our study of the Semantic Web community, the task is to associate scientists with research topics that have been collected manually from the proceedings of ISWC conference series. The system calculates the strength of association between the name of a given person and a certain topic. This strength is determined by taking the number of the pages where the name of an interest and the name of a person co-occur divided by the total number of pages about the person. We assign the expertise to an individual if this value is at least one standard deviation higher than the mean of the values obtained for the same concept.[7] We also borrow from the work of Mutschke and Quan Haase, who perform network analysis based on bibliographic records that contain keywords of publications. Before applying an analysis of the social-cognitive network of co-authors, the authors cluster keywords into themes based on the co-occurrences of keywords on publications, assign documents to themes and subsequently determine which themes are relevant for

[7]Note that we do not factor in the number of pages related to the concept, since we are only interested in the expertise of the individual relative to himself. By normalizing with the page count of the interest the measure would assign a relatively high score —and an overly large number of interests— to individuals with many pages on the Web. We only have to be careful in that we cannot compare the association strength across interests. However, this is not necessary for our purposes.

a person based on his or her publications [Mutschke and Haase, 2001]. We also perform a simple clustering of keywords based on their co-occurrences among the interests of researchers (see 7.5).

Kautz already notes that the biggest technical challenge in social network mining is the disambiguation of person names. Persons names exhibit the same problems of polysemy and synonymy that we have seen in the general case of web search. Queries for researchers who commonly use different variations of their name (e.g. *Jim Hendler* vs. *James Hendler*) or whose names contain international characters (e.g. *Jérôme Euzenat*) may return only a partial set of all relevant documents known to the search engine. Queries for persons with common names such as *Martin Frank* or *Li Ding* return pages about all persons with the same name. Another problem is that the coverage of the Web can be very skewed: for example, *George Bush* the president is over-represented compared to *George Bush* the beer brewer. Not only statistical methods suffer, but also content analysis as in this case the top pages returned by the search engine may not even mention the beer brewer (web pages are largely ranked by popularity). This is a typical web scale problem: such name collisions are rare in even the largest of corporate settings but a common phenomenon on the Web.

There have been several approaches to deal with name ambiguity. Bekkerman and McCallum deal with this problem by using limited background knowledge: instead of a single name they assume to have a list of names related to each other [Bekkerman and McCallum, 2005]. They disambiguate the appearances by clustering the combined results returned by the search engine for the individual names. The clustering can be based on various networks between the returned webpages, e.g. based on hyperlinks between the pages, common links or similarity in content. Bollegala, Matsuo and Ishizuka also apply clustering based on the content similarity but go a step further in mining the resulting clusters for key phrases [Bollegala et al., 2006]. The idea is that such key phrases can be added to the search query to reduce the set of results to those related to the given target individual. For example, when searching for *George Bush* the beer brewer one would add the term *beer* to the query.

In our work in extracting information about the Semantic Web community we also add a disambiguation term our queries. We use a fixed disambiguation term (*Semantic Web OR ontology*) instead of a different disambiguation term for every name. This is a safe (and even desirable) limitation of the query as we are only interested in relations in the Semantic Web context. The method of Bollegala et al. would likely suggest more specific key phrases for every individual and that would increase the precision of our queries, but likely result in much lower recall. (As the co-occurrences we are looking for are relatively rare we cannot afford to lower recall by adding too many or too specific terms.)

We also experiment with a second method based on the concept of *average precision*. When computing the weight of a directed link between two persons we consider an ordered list of pages for the first person and a set of pages for the second (the relevant set) as shown in Figure 4.5. In practice, we ask the search engine for the top N pages for both persons but in the case of the second person the order is irrelevant for the computation. Let's define $rel(n)$ as the relevance at position n, where $rel(n)$ is 1 if the document at

position n is the relevant set and zero otherwise ($1 \leq n \leq N$). Let $P(n)$ denote the precision at position n (also known as *p@n*):

$$P(n) = \frac{\sum_{r=1}^{n} rel(r)}{n}$$

Average precision is defined as the average of the precision at all relevant positions:

$$P_{ave} = \frac{\sum_{r=1}^{N} P(r) * rel(r)}{N}$$

The average precision method is more sophisticated in that it takes into account the order in which the search engine returns document for a person: it assumes that names of other persons that occur closer to the top of the list represent more important contacts than names that occur in pages at the bottom of the list. The method is also more scalable as it requires only to download the list of top ranking pages once for each author. (This makes it linear in the number of queries that need to be made instead of quadratic.) The drawback of this method is that most search engines limit the number of pages returned to at most a thousand. In case a person and his contacts have significantly more pages than that it is likely that some of the pages for some the alters will not occur among the top ranking pages.

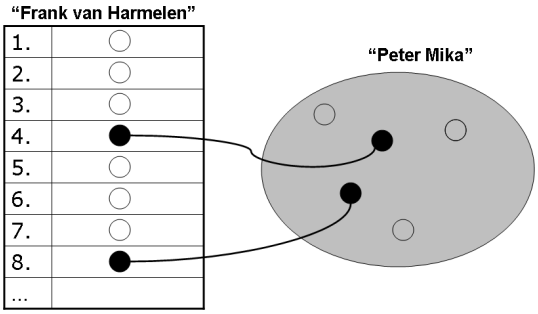


Figure 4.5: The average precision method considers also the position of the pages related to a second person in the list of results for the first person.

Lastly, we would note that one may reasonably argue against the above methods on the basis that a single link or co-occurrence is hardly evidence for any relationship. In fact, not all links are equally important nor every co-occurrence is intended. For example, it may very well happen that two names co-occur on a web page without much meaning to it (for example, they are on the same page of the corporate phone book or appear in a list of citations). What is important to realize about these methods is that they are statistical and assume that the effects of uneven weights and spurious occurrences disappear by means of large numbers. We will evaluate the robustness of both the simple co-occurrence and the average precision based methods in Chapter 8.

4.4 Discussion

The term *e-social science* covers the new movement in Social Sciences toward the (re)use of electronic data for social studies. The basis of this new opportunity for the Social Sciences is the through and through digitalization of modern life, including a transition of much of our communication and social interaction into the online world. The electronic records of online social activity offer a tremendous possibility for social scientist to observe communicative and social patterns at much larger scales than before. (Parallel advances in computer technology make it also possible to deal with the exploding amounts of data at hand.) Further, as the unscalable human element is removed from the data collection process it becomes possible to sample data at arbitrary frequencies and to observe dynamics in a much more reliable fashion. Longitudinal studies built on observation or survey data typically rely on looking at the state of the network at just two distinct time points; the human effort involved in collecting more data points is usually prohibitive.

The expansion of the online universe also means a growing variety of social settings that can be studied online. We have shown above a cross-section of studies in e-social science that represent the diversity of topics in this emerging field. These studies rely on very different data sources, each with its own method of extraction and limitations as to what extent it can be used as a proxy for observing social phenomena.

The method we will use in our work is social network extraction from the content of Web pages. These Information Retrieval based methods have the advantage of generality: they can be used to extract networks of a variety of social units (individuals, institutes etc.) but can also be limited to particular settings by focusing the extraction to certain parts of the Web. The only requirement is that the network to be studied should have a significant coverage on the Web so that the extraction can produce reliable results even if the face of noise e.g. spurious co-occurrences of names on web pages. In Chapter 8 we will evaluate this method and show that it is robust to such noise and suited for network analysis.

The most important technical hurdle to overcome when applying such methods is the problem of person name disambiguation, i.e. the matching of references to actual persons they represent. Person names are neither unique nor single: a person may have different ways of writing his name and two persons may have the same name. Interestingly this is exactly the kind of problem the Semantic Web is designed to solve. Websites such as LiveJournal that already export data in a semantic format do not suffer from this problem. In their case, a machine processable description is attached to every web page, containing the same information that appears on the web page but in a way that individuals are uniquely identified. We will show how such semantic descriptions work in the following Chapter. In Chapter 6 we will apply this technology to a related problem: the merging of multiple electronic data sources for analysis.

Chapter 5

Ontology-based Knowledge Representation

This section gives a brief, non-exhaustive introduction to knowledge representation (modelling) using ontology languages such as RDF and OWL. We explain RDF/OWL by comparing them to related, well-known information modelling techniques familiar to Computer Scientists and practitioners: the E/R and relational models, UML class models and XML. Software engineers and web developers often find RDF and OWL difficult to learn, which has to do with the grounding of these languages in formal logic, which is not well covered in most Computer Science curricula. However, we would like to show that RDF and OWL can also be quickly understood when explained using the more familiar concepts of software and database design. In effect, all these models are based on simple statements about objects, properties and their values. Some basic understanding of logic will only be required when we discuss the notion of semantics in Section 5.1.1.

For authoritative information on RDF and OWL we refer the reader to the appropriate W3C specifications and the more easily readable, non-normative RDF Primer and OWL Guide documents [Manola and Miller, 2004, Smith et al., 2004][1] . In the past few years a number of academic books have also appeared based on research and teaching experience at leading Semantic Web research institutes and universities, e.g. the recommended Semantic Web Primer by Antoniou and Harmelen [Antoniou and van Harmelen, 2004]. Lastly, practical programming guides are also starting to appear as technology becomes more widely adopted [Powers, 2003].

[1]The home page `http://www.w3.org/2001/sw/` of the W3C Semantic Web Activity gives access to a wealth of information, including presentations, articles as well as the technical specifications we reference in this Chapter.

5.1 The Resource Description Framework (RDF) and RDF Schema

The Resource Description Framework (RDF) was originally created to describe resources on the World Wide Web (in particular web pages and other content), hence the name. In reality, RDF is domain-independent and can be used to model both real world objects and information resources. RDF itself is a very primitive modelling language, but it is the basis of more complex languages such as OWL.

There are two kinds of primitives in RDF: resources and literals (character sequences). The definition of a resource is intentionally vague; in general everything is modelled as a resource that can be (potentially) identified and described. Resources are either identified by a URI or left blank. URIs are identifiers with a special syntax defined in [Berners-Lee et al., 1998].[2] Blank resources (blank nodes) are the existential quantifiers of the language: they are resources with an identity, but whose identifier is not known or irrelevant. (These is no way in the RDF model itself to assign an identifier to a blank node.) Literals are strings (character literals) with optional language and datatype identifiers.

Expressions are formed by making statements (triples) of the form (subject, predicate, object). The subject of a statement must be a resource (blank or with a URI), the predicate must be a URI and the object can be either kind of resource or a literal. Literals are thus only allowed at the end of a statement.[3]

RDF is very easy to understand in practice. The following brief RDF document describes a person named Rembrandt (see Figure 5.1). The six statements are also shown as a directed, labelled graph in Figure 5.2. In this visualization the nodes are the subjects and objects of statements, labelled with the URI or literal or left blank, and the edges connect the subjects and objects of statements and are labelled with the URI of the property.[4] As we can see the statements of the RDF model form a graph because the object of one statement can be the subject of another statement. (As noted literals cannot be the subjects of statements, i.e. there are no arrows going from literals.)

This fragment is written in the Turtle language, one of the many syntaxes of RDF. Turtle allows to abbreviate URIs using namespaces; namespaces are not a feature of RDF, but rather syntactic sugar in many RDF serializations. The resources from the FOAF namespace that we use are defined separately in the so called Friend-of-a-Friend (FOAF) ontology, which resides at `http://xmlns.com/foaf/0.1/index.rdf`. We

[2] URLs used on the Web (such as the ones starting with http:) are also a form of URIs. The registration of new URI schemes is regulated in [Petke and King, 1999]. In practice, it is most common to use http: URIs as identifiers even if the resources modelled are not Web resources.

[3] There has been some discussions in the RDF community to allow statements about literals, which would be useful, for example, to define inverses of properties with literal values. For example, in the context of the example in Figure 5.1 this would allow to make statements such as "Rembrandt" *ex*:*firstNameOf ex*:*Rembrandt*. However, it would also encourage poor modelling, in particular confusing resources with their labels (e.g. "Rembrandt" *foaf*:*gender* "male" instead of *ex*:*Rembrandt foaf*:*gender* "male").

[4] This visualization follows the conventions of the RDF Primer document [Manola and Miller, 2004]. Unlike in the case of visual modelling languages such as UML, visualization plays a minor role in RDF modelling. In fact, there is no standard visual modelling language for ontologies; the kind of generic graph visualization shown here results in a very complex image for any real ontology.

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#label> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix example: <http://www.example.org/> .

example:Rembrandt rdf:type foaf:Person .
example:Saskia  rdf:type foaf:Person .
example:Rembrandt foaf:name "Rembrandt" .
example:Rembrandt foaf:mbox <mailto:rembrandt@example.org> .
example:Rembrandt foaf:knows example:Saskia .
example:Saskia  foaf:name "Saskia" .
```

Figure 5.1: A set of triples describing two persons represented in the Turtle language.

discuss FOAF in more detail in Section 6.2. For now it is enough to know that it is another RDF document on the Web defining some of the terms that we are using in the example.

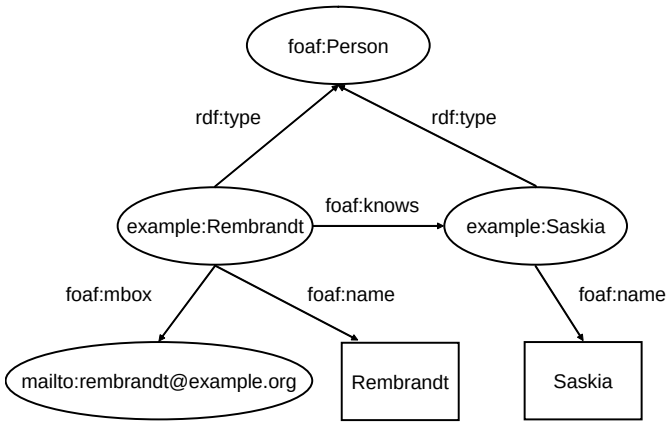


Figure 5.2: A graph visualization of the RDF document shown in Figure 5.1.

RDF is so minimal —in fact we just summarized it in two paragraphs— that in practice it is always used in combination with RDF Schema.[5] RDF Schema is a simple extension of RDF defining a modelling vocabulary with notions such as classes and properties (see Table 5.2). Examples of properties include the subclass property holding between classes and the domain/range properties specifying the domain and range of properties.

[5] In fact, when mentioning RDF most people refer to RDF used in conjunction with RDF Schema. In the following, we will also use the term RDF in this sense and apply the term RDF Schema when we specifically intend to speak about the RDF Schema vocabulary.

The *rdf*:*type* property we have seen above specifies that a resource is an instance of a class.

Figure 5.3 shows some of the statements from the FOAF ontology. The first group of statements describe the class Person. The type of the resource is specified as *owl*:*Class*, we give a label, name a superclass and state that this class is disjoint from the class of documents. (We return to these terms from the OWL vocabulary in Section 5.2). We then describe the *foaf*:*knows* and *foaf*:*name* properties we have used above, specifying their type label, domain, range and a superproperty in the case of *foaf*:*name*.

What is likely to be surprising here is that in comparison to most other modelling frameworks we use the same model of subject/predicate/object to describe instances as well as classes. The statements describing a set of instances (Figure 5.1) and the statements describing the higher level of classes (Figure 5.3) are typically stored in separate documents on the Web, but this is not a requirement: when put together they form a single graph like the one showed in Figure 5.2.

In fact it is not always trivial to separate the description of the instances from the description of classes. We may want to store instance data and the ontology together for practical reasons. But also, for example, we may want to describe a class using typical instances such as when describing the concept of a Weekday, which is a class of five days. In this case drawing a clear line between instance data (metadata) and class data (schema) is problematic even from a conceptual point of view. This is reflected in the rather ambiguous usage of the term ontology, which is sometimes used to refer to the classes only, and sometimes to a set of instances and classes bundled together.

The example also reveals some of the impressive flexibility of RDF. The constructs of language itself form a vocabulary just like the terms of any domain ontology. In other words, terms like *rdfs*:*Class* are not treated any special. This means that it is very well possible to form statements about the elements of the language itself. This is used sometimes to establish relationship between elements of a domain and the RDF Schema vocabulary, for example to state that a domain specific labelling property (such as a property providing the name for a person) is a subproperty of the more general *rdfs*:*label* property. We have done that when declaring the *foaf*:*name* property as a subproperty of the *rdfs*:*label* property. Although it makes little sense, one might even state for example that *rdfs*:*Resource* is an *rdfs*:*subClassOf rdfs*:*Class*, effectively claiming that all resources are classes in the model.

Further, RDF makes no clear separation between classes, instances and properties. One can create classes of instances (metamodeling), which is often required in modelling practical knowledge. (The classical example is modelling the notion of species as a class of classes of animals.) Metamodeling is also present in the example, as we have seen that the *rdf*:*type* is used on both an instance (Rembrandt, whose class is Person), and a class (Person, whose class is the class of all classes) and we used the *rdfs*:*label* property on both instances, classes and properties. Classes of properties are also often used to characterize properties. For example, in the case of the OWL language the resource *owl*:*DataTypeProperty* is the class of properties that have Literal values.

Lastly, part of the flexibility of RDF is that language constructs are not interpreted as strictly as we might expect. For example, even though the range of the *foaf*:*knows* property is defined to be *foaf*:*Person* we can still add the statement that *example*:*Rembrandt*

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

foaf:Person rdf:type owl:Class .
foaf:Person rdfs:label "Person" .
foaf:Person rdfs:subClassOf foaf:Agent
foaf:Person owl:disjointWith foaf:Document .

foaf:knows rdf:type owl:ObjectProperty .
foaf:knows rdfs:label "knows" .
foaf:knows rdfs:domain foaf:Person .
foaf:knows rdfs:range foaf:Person .

foaf:name rdf:type owl:DatatypeProperty .
foaf:name rdfs:label "name" .
foaf:name rdfs:subPropertyOf rdfs:label .
foaf:name rdfs:domain owl:Thing .
foaf:name rdfs:range rdfs:Literal
```

Figure 5.3: Some statements from the FOAF ontology about the terms used in the previous example.

foaf:*knows mailto*:*saskia@example.org*. While one would might expect that this leads to a logical contradiction as we are confusing people and their email addresses, this RDF statement is merely interpreted as a hint that the resource *mailto*:*saskia@example.org* is both an email address *and* a Person.[6] This example points to the importance of understanding the precise meaning of RDF(S) constructs, which is the subject of the following Section.

5.1.1 RDF and the notion of semantics

In the above we have been largely concerned with the syntax of the language: the kind of symbols (resources and literals) we have and the way to form statements from them. Further, we introduced some special resources that are part of the RDF and RDF Schema languages (e.g. *rdfs*:*subClassOf*) and gave their meaning in a colloquial style. However, if we want to use RDF(S) for conveying knowledge across the Web, we need to be able to define the meaning of our language (and thus the meaning of ontologies) in a much more reliable fashion. This is important from the perspective of machine reasoning: for example, we are often interested to check whether a certain statement necessarily follows from a set of existing statements or whether there is contradiction among a set of statements. What we then need is an agreement on a method to unambiguously answer

[6] There is no way of expressing in RDF(S) that such an intersection of classes is empty, but it is possible in OWL, see Section 5.2.

such questions independently of our interpretation of the natural language description of the RDF(S) constructs.

The meaning of RDF(S) constructs is anchored in a model-theoretic semantics, one of the most common ways to provide semantics [Hayes, 2004]. Using model-theoretic semantics meaning is defined by establishing a mapping from one model to a meta-model where the truth of propositions is already uniquely determined. In the context of RDF such a mapping is called an *interpretation*.

Although a meta-model can be defined in another formal system, it is convenient to think of the meta model as some independently existing reality that the ontology is intended to describe. Thus an interpretation can be thought of as a mapping between symbols and the objects or relations they intended to describe (see Figure 5.4).

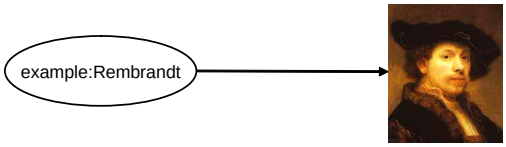


Figure 5.4: An interpretation is a mapping between the terms of an ontology and an interpretation domain.

The constructs of RDF are used to put constraints on possible interpretations and to exclude thereby some of the unintended interpretations. For example, we only want to allow interpretations where symbols for the two persons are mapped to different objects. In other words, we want to exclude interpretations where the two instances are mapped to the same object. In order to achieve this we can specify that the instance Rembrandt is *owl*:*differentFrom* the second instance Saskia. This excludes the interpretation shown in Figure 5.5.

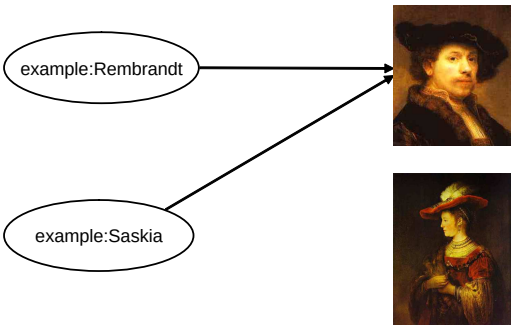


Figure 5.5: An unintended model where symbols for different persons are mapped to the same object. We can exclude this interpretation by adding a statement to the ontology claiming that the two individuals are different from each other. (*owl*:*differentFrom*)

The model theory of RDF(S) provides some axiomatic triples that are true in every RDF(S) interpretation (for example, *rdf*:*Property rdf*:*type rdfs*:*Class*) and defines the constraints that language elements put on possible interpretations of RDF(S) models. In practice, these semantic conditions can be expressed as a set of rules: for example, the semantics of *rdfs*:*subPropertyOf* is given by the following rules:[7]

$$(aaa, \textit{rdfs:subPropertyOf}, bbb) \wedge (bbb, \textit{rdfs:subPropertyOf}, ccc)$$
$$\rightarrow (aaa, \textit{rdfs:subPropertyOf}, ccc)$$
$$(aaa, \textit{rdfs:subPropertyOf}, bbb) \rightarrow (aaa, \textit{rdf:type}, \textit{rdf:Property})$$
$$(aaa, \textit{rdfs:subPropertyOf}, bbb) \rightarrow (bbb, \textit{rdf:type}, \textit{rdf:Property})$$
$$(xxx, aaa, yyy) \wedge (aaa, \textit{rdfs:subPropertyOf}, bbb) \rightarrow (xxx, bbb, yyy)$$

The most noteworthy feature of RDF semantics is that the interpretation of the language is based on an open world assumption and is kept monotonic. An open world assumption means that based on a single document we cannot assume that we have a complete description of the world or even the resources explicitly described therein. Monotonicity means additional knowledge added to an RDF knowledge base cannot make previous inferences invalid.[8] For example, if we specify that the range of the foaf:knows property is Person and then state that Rembrandt knows an instance of another class such as Pluto the dog (or even a literal value) we do not cause a (logical) contradiction; it is assumed that there could exist a statement defining that some other class (e.g. Dog) is also in the range of *foaf*:*knows*.

A consequence of monotonicity is that it makes no sense of talking about RDF validation: RDF Schema statements about an RDF resource only add additional information about the resource and cannot invalidate or contradict statements previously inferred about that resource. In fact, RDF Schema semantics is specified as a set of inference rules; in the example, these rules would allow to infer that the resource provided as the object of the *foaf*:*knows* property is (also) a Person besides possibly being other things (a Dog).[9]

5.2 The Web Ontology Language (OWL)

The Web Ontology Language (OWL) was designed to add the constructs of Description Logics (DL) to RDF, significantly extending the expressiveness of RDF Schema both in characterizing classes and properties. Description Logics are a set of Knowledge Representation languages with formal semantics based on their mapping to First Order Logic

[7] Strictly speaking, the semantics of the property is of course related to the semantics of the *rdf*:*type* property and the *rdf*:*Property* class which occur in these rules. In general, the semantics of symbols is often given using the semantics of other primitive symbols.

[8] It is altogether impossible to create logical inconsistencies in RDF with the exception of a datatype clash.

[9] There is no such thing as a name clash in RDF due to the unique identification of resources and thus it is not possible to conclude that there are two different knows properties or Person classes involved.

(FOL). Description Logics have been extensively studied since the 1980s including studies on the tradeoffs between the expressivity of the chosen language and the efficiency of reasoning. OWL has been designed in a way that it maps to a well-known Description Logic with tractable reasoning algorithms.

The Web Ontology Language is in fact a set of three languages with increasing expressiveness: OWL Lite, OWL DL and OWL Full. These languages are extensions of each other ($OWL_{Lite} \subseteq OWL_{DL} \subseteq OWL_{Full}$) both syntactically and semantically. For example, every OWL Lite document is a valid OWL DL document and has the same semantics when considered as an OWL DL document, e.g. it leads to the same logical conclusions. The vocabularies of these languages extend each other and languages further up in the hierarchy only relax the constraints on the use of the vocabulary. Although it is generally believed that languages of the OWL family would be an extension of RDF(S) in the same sense, this is only true for OWL Full, the most expressive of the family ($RDF(S) \subseteq OWL_{Full}$).

The middle language, OWL DL was the original target of standardization and it is a direct mapping to an expressive Description Logic. This has the advantage that OWL DL documents can be directly consumed by most DL reasoners to perform inference and consistency checking. The constructs of OWL DL are also familiar, although some of the semantics can be surprising mostly due to the open world assumption [Rector et al., 2004]. (Table 5.2 shows the OWL DL vocabulary, which is the same as the vocabulary of OWL Full.) Description Logics do not allow much of the representation flexibility introduced above (e.g. treating classes as instances or defining classes of properties) and therefore not all RDF documents are valid OWL DL documents and even the usage of OWL terms is limited.

For example, in OWL DL it is not allowed to extend constructs of the language, i.e. the concepts in the RDF, RDF Schema and OWL namespaces. In the case of the notion of a Class, OWL also introduces a separate *owl*:*Class* concept as a subclass of *rdfs*:*Class* in order to clearly distinguish its more limited notion of a class. Similarly, OWL introduces the disjoint classes of object properties and datatype properties. The first refers to properties that take resources as values (such as *foaf*:*knows*) and the latter is for properties ranging on literals such as *foaf*:*name*.

OWL Full is a "limitless" OWL DL: every RDF ontology is also a valid OWL Full ontology and has the same semantics when considered as an OWL Full document. However, OWL Full is undecidable, which means that in the worst case OWL Full reasoners will run infinitely. OWL Lite is a lightweight sub-language of OWL DL, which maps to a less expressive but even more efficient DL language. OWL Lite has the same limitations on the use of RDF as OWL DL and does not contain some of the terms of OWL DL.

In summary, RDF documents are not necessarily valid OWL Lite or OWL DL ontologies despite the common conviction (see also Figure 5.9). In fact, "downgrading" a typical RDF or OWL Full ontology to OWL DL is a tedious engineering task. It typically includes many simple steps such as declaring whether properties are object properties or datatype properties and importing the external ontologies used in the document, which

Basic constructs	*rdf*:*type rdf*:*Property rdf*:*XMLLiteral*
Collections	*rdf*:*List rdf*:*Seq rdf*:*Bag rdf*:*Alt rdf*:*first rdf*:*rest rdf*:*nil rdf*:*_1 rdf*:*_2 . . . rdf*:*value*
Reification	*rdf*:*Statement rdf*:*subject rdf*:*predicate rdf*:*object*

Table 5.1: The RDF vocabulary.

Basic constructs	*rdfs*:*domain rdfs*:*range rdfs*:*Resource rdfs*:*Literal rdfs*:*Datatype rdfs*:*Class rdfs*:*subClassOf rdfs*:*subPropertyOf*
Collections	*rdfs*:*member rdfs*:*Container rdfs*:*ContainerMembershipProperty*
Documentation & reference	*rdfs*:*comment rdfs*:*seeAlso rdfs*:*isDefinedBy rdfs*:*label*

Table 5.2: The RDF Schema vocabulary.

is mandatory in OWL but not in RDF. However, the process often involves more fundamental modelling decisions when it comes to finding alternate representations.[10]

Most existing web ontologies make little use of OWL due to their limited needs, but also because general rule-based knowledge cannot be expressed in OWL. The additional expressivity of OWL, however, is required for modelling complex domains such as medicine or engineering, especially in supporting classification tasks where we need to determine the place of a class in the class hierarchy based on its description.

5.3 Comparison to the Unified Modelling Language (UML)

UML is most commonly used in the requirements specification and design of object-oriented software in the middle tier of enterprise applications [Fowler, 2003]. The chief difference between UML and RDF(S)/OWL is their modelling scope: UML contains modelling primitives specific for a special kind of information resource, namely objects in an information system characterized by their static attributes and associations, but also their dynamic behavior. Many of the modelling primitives of UML are thus specific to objects and their role in OO systems; interfaces, functions etc. are examples of such constructs.[11] Nevertheless, if we ignore these constructs of the languages and the differ-

[10] The author has carried out this process both with the OWL-S and FOAF ontologies and reported his experiences on the appropriate mailing lists.

[11] The distinction between modelling information resources and real world objects is not explicit in UML. Some models of UML such as activity diagrams are primarily used for modelling used real world concepts, e.g. business processes. Other diagrams such as use case diagrams indicate how real world actors interact with the system, i.e. they contain real world objects and information resources in the same diagram.

Class Axioms	*owl*:*oneOf, dataRange owl*:*disjointWith owl*:*equivalentClass*
Boolean Combinations	*owl*:*unionOf owl*:*complementOf owl*:*intersectionOf*
(In)Equality	*owl*:*equivalentClass owl*:*equivalentProperty owl*:*sameAs owl*:*differentFrom owl*:*AllDifferent owl*:*distinctMembers*
Property Characteristics	*owl*:*ObjectProperty owl*:*DatatypeProperty owl*:*inverseOf owl*:*TransitiveProperty owl*:*SymmetricProperty owl*:*FunctionalProperty owl*:*InverseFunctionalProperty*
Property Restrictions	*owl*:*Restriction owl*:*onProperty owl*:*allValuesFrom owl*:*someValuesFrom owl*:*hasValue*
Restricted Cardinality	*owl*:*minCardinality owl*:*maxCardinality owl*:*cardinality*
Versioning	*owl*:*versionInfo owl*:*priorVersion owl*:*backwardCompatibleWith owl*:*incompatibleWith owl*:*DeprecatedClass owl*:*DeprecatedProperty*
Ontology annotation	*owl*:*Ontology owl*:*imports owl*:*AnnotationProperty owl*:*OntologyProperty*

Table 5.3: The OWL Full vocabulary.

ence in scope, there is still a significant overlap in the expressiveness of object-oriented models of a domain and ontological models.

Figure 5.6 shows our example modelled in UML. It is natural to model properties that take primitive values as datatypes and model all other properties as associations. (However, attributes can also take model classes as types.) UML is primary a schema definition language and thus the modelling of instances is limited.

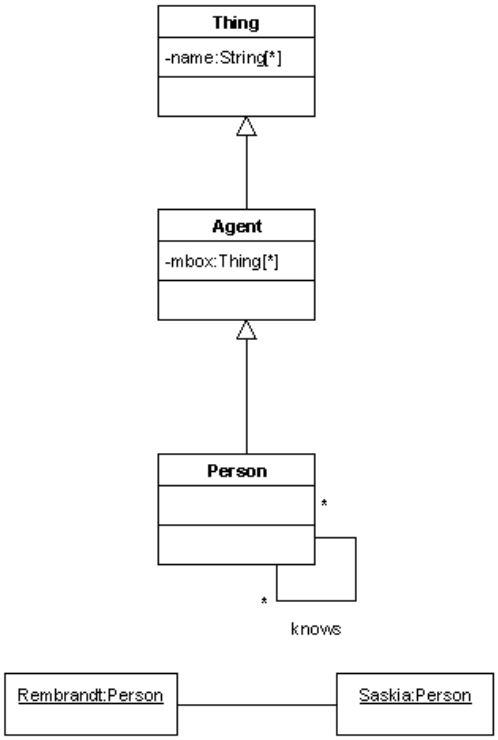


Figure 5.6: A UML model of our example.

Based on the comparison of OWL Full and UML 2.0 we can note that there is a significant overlap as well as differences in the modelling capabilities of OWL and UML [Hart et al., 2004]. In the following we summarize the more specific differences by looking at the unique features of these frameworks.

- **Unique features of RDF/OWL**
  - In general, the modelling of RDF is less constrained than that of UML, which means that many RDF models have no equivalent in UML. OWL DL also provides more primitives than UML such as the disjointness, union, intersection and equivalence of classes.

  - OWL allows to describe defined classes, i.e. definitions that give necessary and sufficient conditions for an instance to be considered as a member of the class.
  - RDF/OWL both treat properties as first class citizens of the language. Properties are global: they do not belong to any class, while UML attributes and associations are defined as part of the description of a certain class. In other words, the same property can be used with multiple classes. Further, RDF properties, just as any other resource can be supplied as values of properties etc.
  - Properties can be defined as subproperties of other properties. This is possible, but much less straightforward in UML.
  - Classes can be treated as instances, allowing for meta-modelling.
  - RDF reification is more flexible than the association class mechanism of UML. For example, statements concerning literal values can also be reified in RDF. These would be modelled as attributes in UML and association classes cannot be attached to attributes.
  - All non-blank RDF resources are identified with a URI, UML classes, instances, attributes etc. do not have such an ID.
  - Instances can and usually have multiple types. (This is not to be confused with multiple inheritance, which is supported by both UML and RDF/OWL.)

- **Unique features of UML**

  - UML has the notion of relationship roles, which is not present in RDF/OWL.
  - UML allows n-ary relations, which are not part of RDF, although they can be re-represented in a number of ways.[12]
  - Two common types of part-whole relations are available in UML (aggregation and composition). These can be remodelled in OWL to some extent[13]
  - UML makes a distinction between attributes and associations. This is also different from the distinction between datatype and object-properties in OWL. On the one hand, attributes can have instances as values, while datatype properties can only have literal values. On the other hand, cardinality constraints, for example, can be applied to both datatype and object properties in OWL, reification can be applied to statements involving datatype properties etc.

A direct comparison of the semantics of UML class models and RDF/OWL models is difficult. The semantics of UML is given by its meta-language (the Meta Object Facility or MOF) and there is no direct mapping between RDF/OWL and the MOF. One obvious difference is that the MOF is built on a closed world assumption while RDF/OWL assumes an open world (see Section 5.1.1).

[12] See the W3C note on n-ary relations at `http://www.w3.org/TR/swbp-n-aryRelations/`

[13] See the W3C note on simple part-whole relations at `http://www.w3.org/2001/sw/BestPractices/OEP/SimplePartWhole/`

As most enterprise software is specified according to OO methodologies and implemented in object oriented languages the relationship of UML and OWL is important to consider. Especially in cases where a Semantic Web application is to be written in an object-oriented framework, it is a natural idea to generate the object model from an existing ontological model or vice versa. There are code generators for creating object models in Java and other languages directly from RDF(S) and OWL ontologies.[14]. Further, there are also tools for the more straightforward conversion from UML class models to OWL ontologies. Lastly, there have been significant work on the creation of a UML profile (presentation syntax) for OWL ontologies called the Ontology Definition Metamodel (ODM)[15]. The ODM will be in effect a visual modelling language for OWL based on UML and the necessary extensions.

5.4 Comparison to the Entity/Relationship (E/R) model and the relational model

The Entity/Relationship (E/R) model is commonly used in information modelling for the data storage layer of applications, because it maps easily to the relational model used for defining data structures in database management systems (RDBMS). It is much simpler than UML, see the original description in [Chen, 1976].

The E/R language contains the constructs that are necessary for modelling information on the basis of relationships. Relationships are characterized in terms of the arity of the relationship and the participating entity sets, their cardinalities and dependency on the relationship. Similar to the reification features in RDF and UML, the E/R model also allows attaching attributes to a relationship and including relationships in other relationships.

Entity sets roughly correspond to classes in UML. (There is no way to represent individual entities.) The modelling of entity sets is less of a concern to the E/R model, the only predefined relationship type between entity sets being generalization. The semantics of this construct is limited to attribute inheritance, i.e. it simply means that the lower level entity sets have all the attributes of the higher level entity sets. (Unlike in UML and RDF, generalization between relationships (*rdfs*:*subPropertyOf*) is not allowed.)

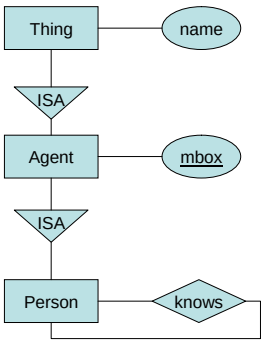
A special feature of the E/R model is the notion of keys of entity sets, i.e. sets of attributes whose values together uniquely identify a given entity. E/R also has a notion of weak entities: entities that are identified not only by a set of attributes but also some of their relations to other entities. (For example the unique identification of a room depends on the identification of the building.) As we will see, keys of single attributes can be modelled in RDF by making the given property functional (max cardinality 1) and inverse functional. Complex keys, however, cannot be accurately modelled in RDF/OWL.

Figure 5.7 shows an E/R representation of our previous example and the corresponding relational model. Notice that the E/R diagram does not allow to represent instances, only the schema of our model. Further, when trying to fill the relational tables with the

[14] See for example Jastor, `http://jastor.sourceforge.net`

[15] `http://codip.grci.com/odm/draft/submission_text/ODMPrelimSubAug04R1.pdf`

instances of our example we run into the problem that instance Saskia is missing the primary key mbox and thus cannot be identified. RDF naturally solves this problem as every resource has an inherent identifier (a URI or bNode ID), which allows to uniquely identify the instance even if the description is partial. (We could have recreated this by introducing a new primary key for the Person entity. This is the equivalent to the practical solution of creating a new column for the Person table with some unique identifier.)



Person

name	mbox
Rembrandt	rembrandt@example.org
Saskia	?

knows

mbox1	mbox2
rembrandt@example.org	?

Figure 5.7: An entity/relationship model of our example, showing also a straightforward mapping to relations, which would be the tables in a database.

Although conceptually the data storage layer of applications is the furthest from their Web interface, the relationship between E/R models and ontologies in an important one to consider. While there are specific storage facilities based on the RDF model (just as there are object databases and relational databases with object extensions), the firm legacy of relational databases means that companies with significant investment in relational databases will most likely to keep their data in an RDBMS and expose it through a web interface in the form of RDF. In such case it makes sense to establish a close mapping between the relational database schema and an ontology that can be published along with the data (http://www.w3.org/DesignIssues/RDB-RDF.html). Unfortunately, the automated mapping of relational schemas to ontologies is difficult in practice. When implementing databases the E/R diagrams may be mapped in multiple ways to the much simpler relational model, which are then often “tweaked” to produce optimal performance. As a result, relational models extracted from existing databases typically require significant post-processing and enrichment to arrive at an accurate model of the domain (see [Volz et al., 2003]).

5.5 Comparison to the Extensible Markup Language (XML) and XML Schema

Up to date XML is the most commonly used technology for the exchange of structured information between systems and services. From all languages discussed the role of XML is thus the most similar to RDF in its purpose.

The most commonly observed similarity between XML and RDF is a similarity between the data models: a directed graph for RDF, and a directed, ordered tree for XML. In XML, the tree is defined by the nesting of elements starting with a single root node. This model originates from the predecessor of XML called SGML which was primarily used for marking up large text documents. Text documents follow the tree structure themselves as paragraphs are nested in subsections, subsections are nested in sections, sections are nested chapters etc. The ordering of the children of an element matters, which is again inherited from the text processing tradition. On the other hand, RDF proposes a more relaxed data model based on arbitrary directed graphs built from single edges between the nodes representing classes or instances. This structure is better suited for creating conceptual domain models, which are often so rich in cross-taxonomical relationships that a hierarchical representation would prove to be difficult. Noting that every tree is a graph (and ignoring the ordering of children for a moment) one would expect that it is trivial to represent XML models in RDF, although as we will see this is not the case.

Much like RDF, XML itself is merely a common conceptual model and syntax for domain specific languages each with their own vocabulary (hence the name extensible). Examples of these languages include XHTML, but also specialized languages such as the GraphML language for descriptions of graphs or the SVG image format for vector graphic images. XML has a number of schema languages such as XML Schema and Relax NG to define such languages. The use of these schema languages, however, is radically different. Namely, schemas written in XML schema languages not only define the types of elements and their attributes but also prescribe syntax i.e. the way elements are allowed to be nested in the tree. XML documents can be validated against a schema on a purely syntactic level. Schema languages for RDF (RDF Schema and OWL) do not impose constraints directly on the graph model but effect the possible interpretations of metadata. Consistency of an RDF/OWL model is checked by searching for possible interpretations. (If there are no possible interpretations of a model than it is inconsistent.) As we have already seen, RDF and OWL also transcend the relatively vague notions of XML (such as elements, attributes, nesting etc.) and provide a more refined set of constructs (classes, instances and properties).

To illustrate the difference in the focus of XML and RDF, let us consider how we would represent the same information captured in Figure 5.1 in XML. Figure 5.8 shows three possible XML representations. While the intended meaning (semantics) of these documents is the same, the syntax and thus the resulting XML tree is different. (The order of the branches matters in XML, which is the only difference between the second and third examples.) This means that applications working on the basis of different representations would be incompatible: XML queries and XSL transformations performed

```
<Person name="Rembrandt">
    <mbox>mailto:rembrandt@example.org</mbox>
    <knows>
        <Person name="Saskia" />
    </knows>
</Person>

<Person name="Rembrandt">
    <mbox>mailto:rembrandt@example.org</mbox>
    <knows>
        <Person ID="1"/>
    </knows>
</Person>
<Person name="Saskia" ID="1" />

<Person name="Saskia" ID="1" />
<Person name="Rembrandt">
    <mbox>mailto:rembrandt@example.org</mbox>
    <knows>
        <Person ID="1"/>
    </knows>
</Person>
```

Figure 5.8: Three different XML representations of the same information.

on these different representations would produce different results. Thus there needs to be an agreement on both syntax and semantics when using XML. For web-based data interchange RDF has a clear advantage here in that agreement on a shared XML format would require a much stronger commitment than the agreement of using RDF. The kind of agreement that is needed to exchange RDF documents concerns only the representation of individual statements (the simple subject/predicate/object model) as we have seen on the example of the Turtle language.

As an aside, we also see the difficulty of XML in dealing with graph-like structures: choosing the top node is an arbitrary decision when dealing with such information. This is particularly obvious in representing social networks (Person entities linked with knows relationships), it is generally true that real world domain models can be rarely forged into a single, unique tree structure.[16]

XML is highly appreciated for the variety of complementary tools and technologies such as XML databases, standard schema, query and transformation languages (XQuery and XSLT), a range of editors, parsers, processors etc. The legacy of XML compelled also the W3C to adopt it also as a notation for RDF and OWL. This kind of contingency is popularized by the famous Semantic Web layer cake diagram in Figure 5.9[17] This

[16] As a fix, XML offers two mechanisms for modelling links between individual elements or branches of the XML tree. One is the simple ID/IDREF mechanism provided by the XML specification, the other is the more sophisticated XLink language for XML.

[17] The diagram has appeared in presentations of the W3C such as `http://www.w3.org/2001/09/06-ecdl/slide17-0.html`.

picture shows the vision of the Semantic Web as a set of languages building upon existing standards such as XML, URIs and Unicode. Despite the good intentions, however, this diagram obfuscates the true relationship between XML and ontology languages from the W3C such as RDF and OWL. (We have already seen that the relationship between RDF and OWL is also more complicated than simple extension.) By doing so it has done more damage to the argument for ontology languages over XML than any other conceptualization of the next generation Web.

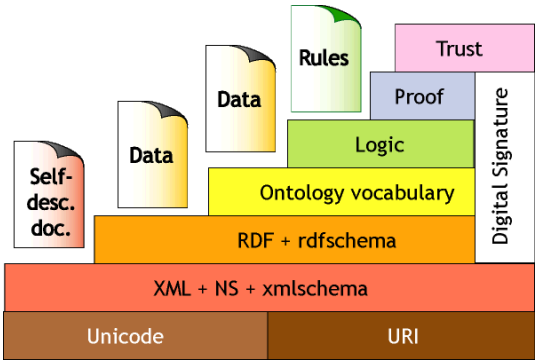


Figure 5.9: The Semantic Web layer cake.

What the layer cake suggests, namely that RDF builds on XML is true only to the extent that RDF models can be exchanged using an XML format named RDF/XML.[18] In the RDF/XML syntax the statements from the graph are simply written out one-by-one with some possibilities for abbreviation. This means that there are a number of possible XML serializations for every RDF graph depending on whether these abbreviations are used, how the statements are ordered etc. In practice, while such RDF/XML documents are valid XML and thus they can be stored in XML databases and queried using XML Query languages such as XQuery or transformed using XSLT, these queries and transformations will be sensitive to a particular serialization. All in all, the only useful tools with respect to RDF/XML files are XML editors and they are only helpful in checking the well-formedness of the representation.

In the other direction, the re-representation of XML schemas and data using RDF is also more difficult than suggested by the layer cake. Transforming XML documents minimally requires assigning namespaces to elements and attributes. Further, RDF/XML mandates a striped syntax requiring that instances and properties are nested correctly into each other. Most XML schemas, however, have not been designed to conform with this striped syntax. For example, the following XML document from the XML Schema Primer violates the assumption of the striped syntax in that it lacks a level of elements between shipTo and name: name is a property an instance of an omitted Person class.

```
<?xml version="1.0"?>
```

[18] While RDF/XML is probably the most common syntax and the one created by the W3C, other notations of RDF such as Turtle, N3 or N-Triples are often preferred over RDF/XML for easier authoring.

```
<apo:purchaseOrder xmlns:apo="http://www.example.com/PO1"
                   orderDate="1999-10-20">
  <shipTo country="US">
    <name>Alice Smith</name>
    <street>123 Maple Street</street>
    <!-- etc. -->
  </shipTo>
  <billTo country="US">
    <name>Robert Smith</name>
    <street>8 Oak Avenue</street>
    <!-- etc. -->
  </billTo>
  <apo:comment>Hurry, my lawn is going wild</apo:comment>
  <!-- etc. -->
</apo:purchaseOrder>
```

While there is no way to automate the process of XML to RDF conversion in the general case, it is possible to design XSLT transformations that carry out this job for particular schemas. The rather inaptly named GRDDL[19] specification provides a standard way of attaching such transformations to XML documents. GRDDL only describes an agreed way to attach transformations to an XML or XHTML document: the agreement on the way to represent and extract metadata is left to the authors of XML formats and transformations. (For example, there is a stylesheet to extract FOAF data from appropriately marked up XHTML documents.[20] Another W3C proposal, the RDF/A syntax gives a generic, domain-independent way to embed RDF metadata in XHTML by adding certain attributes and special elements [Adida and Birbeck, 2006].

For newly designed formats a practical alternative is to create a DTD or XML Schema that takes into account the requirements of RDF/XML or put differently: use RDF/XML in a constrained way so that it conforms to a particular XML Schema. This is the approach that was used in the design of the RSS 1.0 format, which can be meaningfully processed by both XML and RDF tools.

Transforming existing XML Schema documents into RDF/OWL is unfortunately not a trivial task, because the XML model and the rather complex XML Schema standard have notions that RDF/OWL lack and vice versa. For example, XML distinguishes between attributes and terminal elements with simple type values (the difference between the attribute country and the element name in the above example) and there is a natural ordering of elements[21], i.e. the schema for the above example can specify that name has to be listed before email while in RDF the order in which properties are listed cannot be constrained. The advanced features of XML Schema include the definition of uniqueness which is similar to the notion of keys in E/R models. There is also a possibility to define keys and key references linking parts of the schema through an integrity constraint, i.e. that the key reference should contain an existing value from the set of keys. These constructs can not be trivially expressed in OWL either.

[19] GRDDL stands for Gleaning Resource Descriptions from Dialects of Languages. GRDDL is currently a W3C Team Submission found at `http://www.w3.org/TeamSubmission/grddl/`

[20] See `http://www.w3.org/2003/12/rdf-in-xhtml-xslts/grokFOAF.xsl`

[21] except for types defined using the all construct

In summary, RDF/OWL are more flexible, more expressive and conceptually cleaner languages than XML. Their introduction means a conscious break from the XML world on the side of the World Wide Web Consortium. The enormous advantage of XML technology in terms of deployment means however that these modelling paradigms will continue to co-exist for years to come.

5.6 Discussion: Web-based knowledge representation

In the previous section we have introduced RDF and the OWL family of languages. We have compared them to the representations most commonly used today in the three layers of software architectures: E/R models and relational models for database schemas, UML models for object oriented software and middleware and XML schemas for communication and data interchange. In its purpose RDF/OWL are closest to XML. However, they share many of the modelling primitives with E/R and UML class models. The possible mappings between these languages and RDF/OWL is also important to consider when reusing existing data or schemata in a web-based setting.

RDF/OWL have been specifically engineered for knowledge representation in distributed settings such as the Web.

Firstly, RDF/OWL provides the way to distribute data and schema information across the Web and discover such information by traversing the Web of RDF documents. As in the case of XML, RDF documents can reside anywhere on the Web and reference any other RDF vocabulary using globally unique identifiers (URIs). Following the much debated decision of W3C, the use of URIs also connects RDF resources to the Web: URIs with an http: protocol can be looked up on the Web and the response of the Web server can provide a clue whether the resource is an information resource or not.[22] By using redirection, the web server can also provide the authoritative description of the resource. Further, the RDF Schema and OWL vocabularies contain specific terms (*rdfs*:*seeAlso*, *rdfs*:*definedBy* and *owl*:*import*) that can be used to point to other RDF documents to the Web. Using these mechanisms, RDF/OWL documents can be woven into a Web just like HTML documents form a web through the links among the documents.[23]

The second key feature of RDF/OWL is the existence of formal semantics. Formal semantics ascertains that the language is interpreted unambiguously, which is critical to sharing information across the Web. The meaning of the language constructs is defined by the constraints they put on possible mappings to a domain of interpretation (model-theoretic semantics). Informally, the domain of interpretation is convenient to be thought of as some independently existing reality. In this sense the language can be used to exclude unwanted interpretations, i.e. possible mappings to real world objects and relations that we would like to exclude.

[22] See the discussion over the issue named httpRange-14

[23] XML Schema has similar features although they are much less intuitive to use and therefore mostly ignored. instance data can reference the location of a web-based schema through the xsi:schemaLocation attribute, while schemas can reference, import or extend the elements of other web-based schemas. Section 4.3 of the XML Schema specification defines the relationship between XML Schema documents and the Web infrastructure. The use of these advanced schema extension features is not widespread nor recognized as a precursor to Semantic Web techniques.

	Origin	Application domain	Primitive	Expressivity	Distributed representation	Formal semantics
E/R	1976	Relational databases	Relation	•	no	no
UML	1995	OO software	Object	••	no	yes[24]
XML	1998	Text markup and data exchange	Entity	••	yes	no
RDF/OWL	2004	Resource markup and data exchange	Resource	• - • • •	yes	yes

Table 5.4: Comparison of the E/R, UML, XML and RDF/OWL languages.

A clear difference between OWL and RDF is the non-monotonicity of the logic of OWL and the possibility to create inconsistencies. As negation is part of the language it is trivial to create situations where new knowledge contradicts previous assertions. As a result, OWL behaves more like a constraint language, where contradictions in the schema itself (for example, unsatisfiable classes) or the violation of the constraints by the instances (for example, a violation of cardinalities) can be detected as logical inconsistence. OWL reasoners can help to localize the source of inconsistencies, but the automation of removing inconsistencies is still a research problem.

In summary, RDF/OWL are a set of languages of varying expressivity specifically designed for information sharing across the Web. RDF/OWL are different in their purpose from other well-known knowledge representation formalisms used in Computer Science but share many primitives with E/R, UML and XML models. (Table 5.4 summarizes some of our observations.) In particular, accurately representing and sharing the meaning of information (formal semantics) is critical to information sharing in heterogeneous setting where one cannot rely on other mechanisms of creating a shared understanding. A semantic-based representation of knowledge is also an important first step toward data integration in cases where data is distributed and resides under diverse ownership and control. This is the topic of the following Section.

[24] From UML 2.0

Chapter 6

Modelling and aggregating social network data

In this Chapter we apply the ontology-based Knowledge Representation techniques introduced in Chapter 5 to the representation and aggregation social network data. There are two fundamental reasons for developing semantic-based representations of social networks.

Firstly, as we will demonstrate, maintaining the semantics of social network data is crucial for aggregating social network information, especially in heterogeneous environments where the individual sources of data are under diverse control. The benefits are easiest to realize in cases where the data are already available in an electronic format, which is typically the case in the network analysis of online communities but also in the intelligence community.[1]

Secondly, semantical representations can facilitate the exchange and reuse of case study data in the academic field of Social Network Analysis. The possibilities for electronic data exchange has already revolutionized a number of sciences with the most well-known examples of bio-informatics and genetics. With the current state-of-the art in network analysis, however, the network data collected in various studies is stored and published either in data formats not primarily intended for network analysis (such as Excel sheets, SPSS tables) or in the rather proprietary graph description languages of network analysis packages that ignore the semantics of data, e.g. the types of instances, the meaning of the attributes etc. Hence, it is difficult —if not impossible— to verify results independently, to carry out secondary analysis (to reuse data) and to compare results across different studies. This last effect is particularly hurtful because single studies in network analysis are always focused on relatively small communities while the field as a whole tries to explain commonalities across all social networks.

In the following, we first sketch the state-of-the-art in network data representation and motivate the case for semantics. We discuss separately the possibilities for semantic-

[1] Little is known in this respect, but it is widely speculated that U.S. intelligence agencies are major users of both network analysis and data consolidation/data mining technologies.

based representations of social individuals (Section 6.2) and social relationships (Section 6.3). While the representation of social individuals is relatively straightforward, the representation of social relations is a much more challenging problem. Our main contribution is thus a first step towards an ontology of social relations.

While reasoning with social relations is still future work, we discuss the aggregation of social individuals, which can be well automated based on the current representations and their formal semantics (Section 6.4). This is also a key technical component of our end-to-end system for collecting, aggregating and presenting social network data. In the following we discuss the design options for implementing semantics-based data aggregation and return to our particular implementation in the following Chapter.

6.1 State-of-the-art in network data representation

In the chapters before we have seen that the most common kind of social network data can be modelled by a graph where the nodes represent individuals and the edges represent binary social relationships. (Less commonly, higher arity relationships may be represented using hyper-edges, i.e. edges connecting multiple nodes.) Additionally, social network studies build on attributes of nodes and edges, which can be formalized as functions operating on nodes or edges.

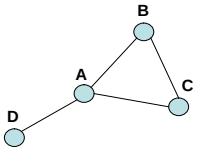
A number of different, proprietary formats exist for serializing such graphs and attribute data in machine-processable electronic documents. The most commonly encountered formats are those used by the popular network analysis packages Pajek and UCINET. These are text-based formats which have been designed in a way so that they can be easily edited using simple text editors. Figure 6.1 shows a simple example of these formats.

Unfortunately, the two formats are incompatible. (UCINET has the ability to read and write the .net format of Pajek, but not vice versa.) Further, researchers in the social sciences often represent their data initially using Microsoft Excel spreadsheets, which can be exported in the simple CSV (Comma Separated Values) format. As this format is not specific to graph structures (it is merely a way to export a table of data), additional constraints need to be put on the content before such a file can processed by graph packages. To further complicate matters for the researcher, visualization software packages also have their own proprietary formats such as the dot format used by the open source GraphViz package developed at AT&T Research.[2]

The GraphML format represents an advancement over the previously mentioned formats in terms of both interoperability and extensibility [Brandes et al., 2004, Brandes et al., 2001]. GraphML originates from the information visualization community where a shared format greatly increases the usability of new visualization methods. GraphML is therefore based on XML with a schema defined in XML Schema. This has the advantage that GraphML files can be edited, stored, queried, transformed etc. using generic XML tools.

Common to all these generic graph representations is that they focus on the graph structure, which is the primary input to network analysis and visualization. Attribute

[2] `http://www.graphviz.org/`




```
*Vertices 4
1 "A"
2 "B"
3 "C"
4 "D"
*Edges
1 1
1 2
1 3
1 4
2 3
```

```
dl
n = 4
labels embedded
format = edgelist
data:
A B
A C
A D
B C
```

```
<?xml version="1.0" encoding="UTF-8"?>
<graphml xmlns="http://graphml.graphdrawing.org/xmlns">
  <graph id="G" edgedefault="undirected">
    <node id="a"/>
    <node id="b"/>
    <node id="c"/>
    <node id="d"/>
    <edge source="a" target="b"/>
    <edge source="a" target="c"/>
    <edge source="a" target="d"/>
    <edge source="b" target="c"/>
  </graph>
</graphml>
```

Figure 6.1: A simple graph (upper left) described in Pajek .NET, UCINET DL and GraphML formats.

data when entered electronic form is typically stored separately from network data in Excel sheets, databases or SPSS tables.[3]

However, none of these formats support the aggregation and reuse of electronic data, which is our primary concern. To motivate our case for data aggregation, consider the typical scenario in Figure 6.2 where we would like to implement a network study by reusing a number of data sources describing the same set of individuals and their relationships (for example, email archives and publication databases holding information about researchers). One of the common reasons to use multiple data sources is to perform *triangulation* i.e. to use a variety of data sources and/or methods of analysis to verify the same conclusion [Robson, 2002]. Often the data sources to be used contain complementary information: for example, in *multi-layer studies* such as the one performed by Besselaar et al. [Heimeriks et al., 2003] a multiplex network is studied using

[3]GraphML goes in this direction by using the extension features of XML Schema. For example, the GraphML node element can be redefined to attach extra attributes to nodes. (These attributes may be defined in other existing schemas.) However, GraphML gives no standard way to define the types of nodes or edges. More precisely: these elements have a type and XML does not allow providing multiple types to an element.

data sources that contain evidence about different kinds of relationships in a community. This allows to look at, for example, how these networks differ and how relationships of one type might effect the building of relationships of another type.

In both cases we need to be able to recognize matching instances in the different data sources and merge the records before we can proceed with the analysis. In fact, the graph representations discussed above strip social network data of exactly those characteristics that one needs to consider when aggregating data or sharing it for reuse: namely, these graph formats reduce social individuals and their relationships to nodes and edges, which is the only information required for the purposes of analyzing single networks.

What we need to support aggregation and reuse is a representation that allows to capture and compare the identity of instances and relationships. (The instances we deal with are primarily persons, but it might be for example that we need to aggregate multiple databases containing patents or publications, which is then used to build a network of individuals or institutions.) Maintaining the identity of individuals and relationships is also crucial for preserving our data sets in a way that best enables their reuse and secondary analysis of data.

Solving these problems requires a very different kind of representation from the graph based formats shown above. Our proposed solution is a rich, semantic-based representation of the primary objects in social networks data, namely social individuals and their relationships. A semantic-based representation will allow us to wield the power of ontology languages and tools in aggregating data sets through domain-specific knowledge about identity, i.e. what it requires for two instances to be considered the same. As we will see, a semantic-based format has the additional advantage that at the same time we can easily enrich our data set with specific domain knowledge such as the relatively simple fact that if two people send emails to each other, they know each other, or that a certain kind of relationship implies (or refutes) the existence of another kind of relationship.

The two key problems in aggregating social network data are the identification and disambiguation of social individuals and the aggregation of information about social relationships. We treat these problems separately in the following Sections.

6.2 Ontological representation of social individuals

The Friend-of-a-Friend (FOAF) ontology that we use in our work is an OWL-based format for representing personal information and an individual's social network. FOAF greatly surpasses graph description languages in expressivity by using the powerful OWL vocabulary to characterize individuals. More importantly, however, using FOAF one can rely on the extendibility of the RDF/OWL representation framework for enhancing the basic ontology with domain-specific knowledge about identity.

The classes and properties in the current version of the FOAF vocabulary are shown in Figure 6.1. We have seen an example of a FOAF profile in Figure 5.1. In terms of deployment, two studies give an insight of how much the individual terms of the FOAF are used on the Web [Paolillo and Wright, 2004, Ding et al., 2005]. Both studies note that the majority of FOAF profiles on the Web are auto-generated by community sites

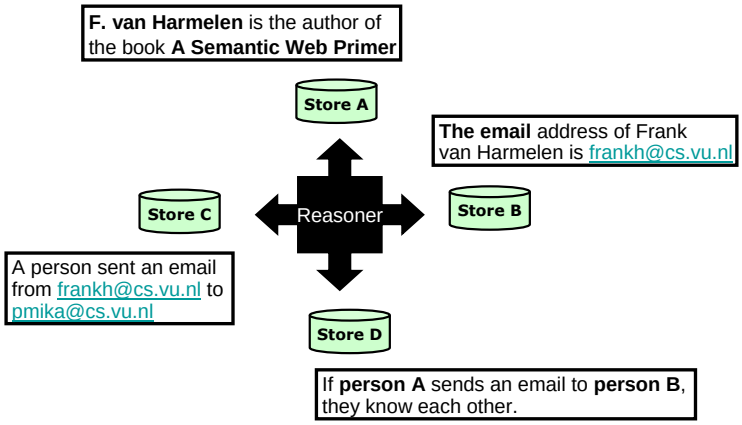


Figure 6.2: Example of a case of identity reasoning. Based on a semantic representation, a reasoner would be able to conclude, for example, that Peter Mika knows the author of the book *A Semantic Web Primer* [Antoniou and van Harmelen, 2004]

such as LiveJournal and Tribe.net. As FOAF profiles are scattered across the Web it is difficult to estimate their number, but even the number of manually maintained profiles is likely to be in the tens of thousands.

FOAF started as an experimentation with Semantic Web technology. The idea of FOAF was to provide a machine processable format for representing the kind of information that made the original Web successful, namely the kind of personal information described in homepages of individuals. Thus FOAF has a vocabulary for describing personal attribute information typically found on homepages such as name and email address of the individual, projects, interests, links to work and school homepage etc. FOAF profiles, however, can also contain a description of the individual's friends using the same vocabulary that is used to describe the individual himself. Even more importantly, FOAF profiles can be linked together to form networks of web-based profiles.

FOAF became the center point of interest in 2003 with the spread of Social Networking Services such Friendster, Orkut, LinkedIn etc. Despite their early popularity, users have later discovered a number of drawbacks to centralized social networking services. First, the information is under the control of the database owner who has an interest in keeping the information bound to the site and is willing to protect the data through technical and legal means. The profiles stored in these systems typically cannot be exported in machine processable formats (or cannot be exported legally) and therefore the data cannot be transferred from one system to the next. (As a result, the data needs to be maintained separately at different services.) Second, centralized systems do not allow users to control the information they provide on their own terms. Although Friendster follow-ups offer several levels of sharing (e.g. public information vs. only for friends), users often still find out the hard way that their information was used in ways that were not intended.

These problems have been addressed with the use of Semantic Web technology. Unlike in the case of Friendster and similar sites, FOAF profiles are created and controlled by the individual user and shared in a distributed fashion. FOAF profiles are typically posted on the personal website of the user and linked from the user's homepage with the HTML META tag. The distributed nature of FOAF networks means that FOAF requires a mechanism to link individual profiles and thus allow the discovery of related profiles. For this purpose, FOAF uses the *rdfs*:*seeAlso* mechanism described above. Related profiles can be thus discovered by crawling the FOAF network along these links.[4]

The distributed nature of FOAF data also means that the designers of the ontology had to address the issues of identification and aggregation early on.[5]. It was very clear from the beginning that again centralized solutions such as a registry for the URIs of individuals would not be feasible. When making a list of friends how would one find out their URIs?

The answer is to identify persons using their characteristic properties such as their email address, homepage etc. When describing a friend one would create a blank node with enough detail to uniquely identify that particular person. Recall that blank nodes in RDF are unidentified resources which can be thought of as the existential quantifier of the language. For example, the statements in the example of Figure 5.1 can be read as *There exists a Person named Peter with email ... who knows a person named Dirk.*.

With the introduction of OWL it has become possible to describe these identifying properties of the *foaf*:*Person* class as inverse-functional properties (IFP). IFPs are properties whose value uniquely identifies a single object.[6] In other words, if two resources have the same value for an inverse-functional property than they must denote the same object. For example, email address is an inverse-functional as every email address belongs to a single person.[7] Name, however is not inverse-functional: the same name can belong to multiple persons. (Nevertheless, the probability of name clashes might be negligibly small depending on the size of the community we consider). We return to the issue of reasoning with this kind of knowledge in Section 6.4.

An advantage of FOAF in terms of sharing FOAF data is the relative stability of the ontology. The number of FOAF users means that the maintainers of the ontology are obliged to keep the vocabulary and its semantics stable. Interestingly, contingency requires that even inconsistencies in the naming of terms are left uncorrected: while most terms follow the Java convention[8], in some cases an underscore is used to separate words (mbox_sha1sum, family_name). When the meaning of terms does change, the evolution

[4]This is the way the so-called scutters (RDF crawlers) operate. Needless to say search engines such as Google that have the capacity of crawling the HTML web will also discover profiles that are linked to individual homepages by either a meta tag or a regular HTML anchor tag. Google is thus also a source of FOAF profiles. The easiest way to locate such profiles is by restricting the search by file type and searching for foaf:Person as a keyword.

[5]In fact, FOAF is the first major ontology where this issue surfaced, see `http://rdfweb.org/mt/foaflog/archives/2003/07/10/12.05.33/index.html`

[6]Mathematically, IFPs are inverses of functional properties and hence the name. For example the inverse of the IFP *foaf*:*mbox* would be a functional property ("mailboxOf").

[7]With the exception of generic email addresses such as *support@microsoft.com*, which belongs to multiple persons. Such email addresses should not be used as values for this property.

[8]`http://java.sun.com/docs/codeconv/html/CodeConventions.doc8.html`

FOAF Basics
Agent
Person
name
nick
title
homepage
mbox
mbox_sha1sum
img
depiction (depicts)
surname
family_name
givenname
firstName

Personal Information
weblog
knows
interest
currentProject
pastProject
plan
based_near
workplaceHomepage
workInfoHomepage
schoolHomepage
topic_interest
publications
geekcode
myersBriggs
dnaChecksum

Online Accounts / IM
OnlineAccount
OnlineChatAccount
OnlineEcommerceAccount
OnlineGamingAccount
holdsAccount
accountServiceHomepage
accountName
icqChatID
msnChatID
aimChatID
jabberID
yahooChatID

Projects and Groups
Project
Organization
Group
member
membershipClass
fundedBy
theme

Documents and Images
Document
Image
PersonalProfileDocument
topic (page)
primaryTopic
tipjar
sha1
made (maker)
thumbnail
logo

Table 6.1: Classes and properties of the FOAF ontology.

is toward generalization so as not to break existing applications.[9] For example, various sub-properties of the *foaf*:*knows* (*foaf*:*knowsWell* and *foaf*:*friend*) term have been removed due to their overlap in meaning. The description of the *foaf*:*schoolHomepage* has been extended to sanction its use for describing university homepages.[10] To facilitate adoption, terms are not added to the vocabulary any more, rather authors are encouraged to create extensions using the mechanisms of RDF, e.g. creating subclasses and subproperties, using the FOAF terms as domains or ranges for other properties etc.

[9] A primary concern in the change management (versioning) of ontologies is the effect of changes on applications. See the thesis of Klein on the issues regarding change management in distributed ontologies [Klein, 2004].

[10] The term school has this broader meaning in the US, while the original authors of the vocabulary are from the UK.

While FOAF has a rich ontology for characterizing individuals —especially with respect to their online presence—, but it is rather poor as a vocabulary for describing relationships. There is a single *foaf*:*knows* relationship defined between Persons and this relationship has no ontological restrictions on its use and is broadly defined in text on the basis of what is not required (e.g. that the relationship is reciprocated or an actual meeting has taken place). This is intentional on behalf of the creators of the vocabulary who wanted FOAF to be applicable in the widest scope possible. However, the consequence is that the meaning of the knows term is now significantly diluted through usage.

The makers of the vocabulary expected, however, that others would use the extensibility of the RDF/OWL language to define more precise notions of relationships. For example, one may define in RDF a relationship called *example*:*supervises* between a *example*:*Teacher* and a *example*:*Student*, where supervises is a subPropertyOf "knows" and Teacher and Student are subclasses of *foaf*:*Person*. OWL would allow to put additional constraints on the use of this relationship, for example to say that every Student is supervised by at least one Teacher. Nevertheless, to characterize relationships beyond the two participants it is desirable to have a representation where relationships themselves are the object of the ontology. This is the subject of the following section.

6.3 Ontological representation of social relationships

Ontological representations of social networks such as FOAF need to be extended with a framework for modelling and characterizing social relationships for two principle reasons: (1) to support the automated integration of social information on a semantical basis and (2) to capture established concepts in Social Network Analysis.

In this section we approach this task using the engineering method of decomposition. The key idea is that the representation of social relationships needs to be fine-grained enough so that we can capture all the detail from the individual sources of information in a way that these can be later recombined and taken as an evidence of a certain relationship.

Network analysis itself can be a help in that it has a rich vocabulary of characterizing social relationships. As these are the terms that are used by social scientists, they are prime candidates to be included in any ontology. For illustration, we list below some of the most commonly discussed characteristics of social relationships. (We specifically focus on interpersonal-relations, ignoring social relationships at different level of analysis, such as institutional relationships or institutional trust.)

- Sign: (*valence*) A relationship can represent both positive and negative attitudes such as like or hate. The positive or negative charge of relationships is important on its own for the study of balance within social networks, which is the subject of balance theory.

- Strength: The notion of tie strength was first introduced by Granovetter in his groundbreaking work on the benefits of weak ties [Granovetter, 1973]. Tie strength itself is a complex construct of several characteristics of social relations. In her

survey, Hite lists the following aspects of tie strength discussed in the literature: Affect/philos/passions , Frequency/frequent contact , Reciprocity, Trust/enforceable trust , Complementarity, Accommodation/adaptation, Indebtedness/imbalance, Collaboration, Transaction investments, Strong history, Fungible skills, Expectations, Social capital, Bounded solidarity, Lower opportunistic behavior, Density, Maximize relationship over org., Fine-grained information transfer, Problem solving, Duration, Multiplexity, Diffusion, Facilitation, Personal involvement, Low formality (few contracts), Connectedness [Hite, 2003].

As this list shows, the conceptualization of tie strength is rather fuzzy. There is little agreement in the field as to the importance of these individual aspects of tie strength [Marsden and Campbell, 1984]. In practice researchers tend to ignore aspects that are irrelevant to their actual study. More unfortunate is the fact that no agreed upon method has emerged yet for measuring them, which means that researchers in the field use different elicitation methods and questions when it comes to measuring any of these aspects. As a result data about tie strength as a numerical value or as a binary distinction between weak and strong ties is hardly comparable across studies. Nevertheless, there is a need for representing measured tie strength, for example, for purposes of secondary analysis.

- Provenance: A social relationship may be viewed differently by the individual participants of the relationship, sometimes even to the degree that the tie is unreciprocated, i.e. perceived by only one member of the dyad. Similarly, outsiders may provide different accounts of the relationship, which is a well-known bias in SNA.

- Relationship history: Social relationships come into existence by some event (in the most generic, philosophical sense) involving two individuals. (Such an event may not require personal contact (e.g. online friendships), but it has to involve social interaction.[11] From this event, social relationships begin a lifecycle of their own during which the characteristics of the relationship may change through interaction or the lack of (see e.g. Hite and Hesterly [Hite and Hesterly, 2001]).

- Relationship roles: A social relationship may have a number of social roles associated with it, which we call relationship roles. For example, in a student/professor relationship within a university setting there is one individual playing the role of professor, while another individual is playing the role of a student. Both the relationship and the roles may be limited in their interpretation and use to a certain social context (see below). Social roles, social contexts and their formalization are discussed in Masolo et al. [Masolo et al., 2004]

In the case of Web-based social networking services we also see a variety of ways of capturing relationship types and other attributes of relationships. For example, Orkut allows to describe the strength of friendship relations on a 5-point scale from “haven’t

[11] Note that the “knows” notion of FOAF is somewhat misleading in this sense, e.g. I know (cognitively recognize) George Bush, but I certainly never had any social interaction with him.

met" to "best friend", while other sites may choose other scales or terms. Further, various sites focus on very different kind of relationships and exchanges, e.g. LinkedIn differs from Orkut in focusing on professional exchanges.

Ideally, all users of all these services would agree to a single shared typology of social relations and shared characterizations of relations. However, this is neither feasible nor necessary. What is required from such a representation is that it is minimal in order to facilitate adoption and that it should preserve key identifying characteristics such as the case of identifying properties for social individuals. Consider that two FOAF-aware software systems can determine if two person objects reference the same individual even if they define different subtypes of the *foaf*:*Person* class.

In summary, a rich ontological characterization of social relationships is needed for the aggregation of social network information that comes from multiple sources and possibly different contexts, which is the typical scenario of the Web but is also the case for network data aggregation in the social sciences. We propose a representation of relationships on the basis of patterns (descriptions) that can be mapped to or extracted from observations about the environment. In other words, we consider relationships as higher-order concepts that capture a set of constraints on the interaction of the individuals.

6.3.1 Conceptual model

The importance of social relationships alone suggests that they should be treated as first-class citizens. (In other words, we would like to discuss social relations as objects, not only as relations between objects.) Alternatively, social relations could be represented as n-ary predicates; however, n-ary relations are not supported directly by the RDF/OWL languages. There are several alternatives to n-ary relations in RDF/OWL described in a document created by the Semantic Web Best Practices and Deployment Working Group of the W3C.[12]

In all cases dealing with n-ary relations we employ the technique that is known as *reification*: we represent the relation as a class, whose instances are concrete relations of that type. One may recall that RDF itself has a reified representation of statements: the *rdf*:*Statement* object represents the class of statements. This class has three properties that correspond to the components of a statement, namely *rdf*:*subject*, *rdf*:*predicate*, *rdf*:*object*. These properties are used to link the statement instance to the resources involved in the statement.

Thus the trivial way to enrich the representation of relations in FOAF is to use the reification feature of the RDF language. We propose two alternative RDF(S) representation of relationships, both using the reification mechanisms of RDF(S) to reify the original triple asserting the existence of the relationship (for example, a *foaf*:*knows* statement). In other words relationships become subclasses of the *rdf*:*Statement* class (see Figures 6.3 and 6.4). Common also to both representations is that the new Relationship class is related to a general Parameter class by the hasParameter relationship. Relationship types such as Friendship are subclasses of the Relationship class, while their parameters (such as strength or frequency) are subtypes of the Parameter class. Note that the

[12] `http://www.w3.org/TR/swbp-n-aryRelations/`

hasParameter metaproperty cannot be defined in OWL DL (its domain is *rdf*:*Statement* while its range is *owl*:*Class* or some subclass of it).

The two alternatives differ in the representation of parameters. The first scheme borrows from the design of OWL-S for representing service parameters, as used in the specification of the profile of a Web Service [Mika et al., 2004]. Here, parameters are related by the valued-by metaproperty to their range (*owl*:*Thing* or a datatype, depending on whether the parameter takes objects or datatypes as values). For example in an application Strength may be a subclass of Parameter valued-by integers. The disadvantage of this solution is that specifying values requires two statements or the introduction of a constructed property (the necessary axiom is not expressible in OWL).

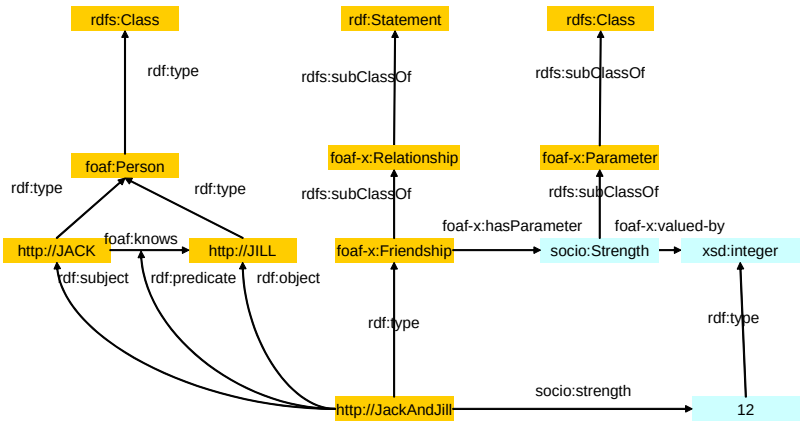


Figure 6.3: An RDF representation of social relationships.

The second alternative differs in that the "native" method of RDF is used for representing parameters: the generic Parameter class is defined as a subclass of *rdf*:*Property*. This model has the advantage that it becomes more natural to represent parameter values and restrictions on them. The disadvantage is that this solution is not compliant with OWL DL: declaring properties ranging on properties and creating subclasses *rdf*:*Property* are not allowed in this species of OWL.

The advantage of describing relations using statement reification is that this form of reification is directly supported by RDF and can be efficiently stored and queried in quad-based triple stores (see Chapter 7).

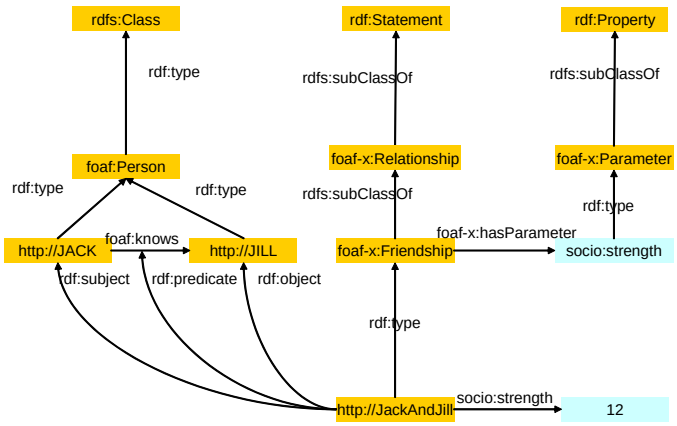


Figure 6.4: An alternative RDF representation of social relationships.

Some caution is also due when using RDF reification. Unlike the reader might expect, reification has no formal semantics in RDF or OWL[13]. In particular, the existence of the reified version of a statement does not imply the existence of the ground statement or vice versa. Further, many would expect reification to be a form of quotation, i.e. the representation of a certain stating of a particular statement that can be attributed to someone. This is not the case due to the interaction with the semantics of OWL.[14] The opposite view, i.e. that two *rdf*:*Statement* instances with the same subject, predicate and object are necessarily identical, does not hold either [Hayes, 2004].

The simple reification based representation is also not powerful enough to capture social relationships with an arity higher than two. For example, brokering is a relationship between two people and a broker. Such relationships cannot be directly represented in RDF and therefore they cannot be reified either.

Further, there are two related characteristics of social relations that we would like to capture in a more advanced representation. The first characteristic is the context dependence of social relations and the second is the separation between an observable situation and its interpretation as a social relation between two individuals.

Social relations are socially constructed objects: they are constructed in social environments by assigning a label to a common pattern of interaction between individuals. Much like social roles discussed in the work of Masolo et al. [Masolo et al., 2004], social

[13]The use of the reification vocabulary is allowed in OWL, i.e. it is not considered an extension of the otherwise protected rdf namespace.

[14]See `http://lists.w3.org/Archives/Public/semantic-web/2006Mar/0194.html`

relationships have a strong contextual dependence in that they own their definition (the ability to identify them) to the social context in which they are interpretable. Only within this social context are we able to identify and interpret a certain interaction-pattern among individuals as a certain kind of relationship. This process, which is known as cognitive structuring, works by applying the generic pattern we associate with such a relationship to the actual state-of-affairs we observe. However, the same observed state-of-affairs may be interpreted according to another pattern as a different kind of relationship. For example, a student/professor relationship at the Free University of Amsterdam (and the attached role of student and professor) is defined by the social context of the university and this kind of relationship may not be recognizable outside of the university. (In another sense, we may talk about student as the entire class of roles of students at learning institutions around the world.)

The individual relationships and their generic description are thus clearly separate. The generic pattern of the relationship comprises those and only those aspects that are shared among particular occurrences of the relationship (for example, there are always two distinct roles in the case of a student/professor relationship with certain requirements for playing those roles). The description is partial in the sense that it allows variation in the particular relations between individuals.

The representation of context and the separation of the level of state-of-affairs (observations of objects and sequences of events) from the higher level of descriptions (contexts) that can be used to interpret those state-of-affairs turns out to be a common problem in the representation of much of human knowledge. A solution proposed by [Gangemi and Mika, 2003] is the Descriptions and Situations ontology design pattern that provides a model of context and allows to clearly delineate these two layers of representation (Figure 6.5).

D&S is a generic pattern for modelling non-physical objects whose intended meaning results from statements, i.e. it emerges in combination with other entities. For example, a norm, a plan, or a social role is usually represented as a set of statements and not as a concept. On the other hand, non-physical objects may change and be manipulated similar to physical entities, and are often treated as first-order objects. That means that an ontology should account for such objects by modelling the context or frame of reference on which they depend.

D&S is an ontology-design pattern in the sense that it is used as a template for creating domain ontologies in complex areas. D & S has been successfully applied in a wide range of real-life ontology engineering projects from representing Service Level Agreements (SLAs) to the descriptions of Web Services [Mika et al., 2004].

D&S builds on some basic categories from the DOLCE foundational ontology, namely the notions of Objects, Events and Regions. (These concepts represent the top level ontological choice in almost all Foundational Ontologies.) As depicted in the Figure, the notion of Context in D & S is composed of a set of Parameters, Functional Roles and Courses of Events. Axioms enforce that each descriptive category acts as a selector on a certain basic category of DOLCE: Parameters are valued-by Regions, Functional Roles are played-by Objects (endurants) and Courses of Events sequence Events (perdurants). The elements of the context thus mirror the elements of the state-of-affairs (a set of objects, events and their locations), but add additional semantics to them. Note

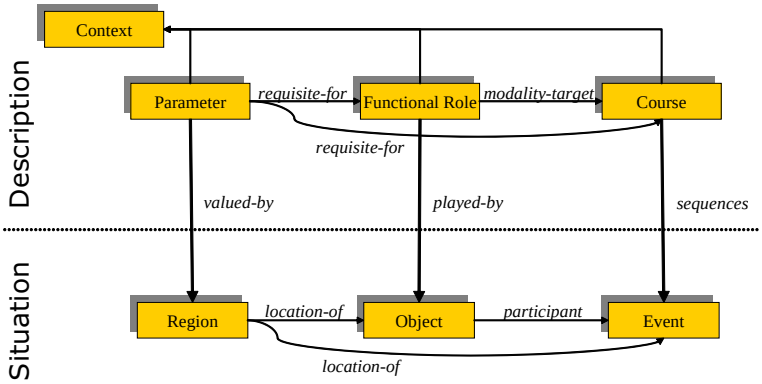


Figure 6.5: The Descriptions and Situations ontology design pattern.

also that these levels of description and situation are clearly separate in that the same state-of-affairs may be interpreted according to another theory by mapping the elements of that other theory to the same set of objects and events. D & S captures the intuition that multiple overlapping (or alternative) contexts may match the same world or model, and that such contexts can have systematic relations among their elements.

D&S has been already used by Masolo et al. for the representation of social contexts and social roles. Their arguments about the context dependence of social roles equally hold for social relations and we follow their approach in using D&S for the design of our conceptual model for the representation of social relationships. In particular, we model a Social Relationship as a subclass of Context and particular social relationships such as Friendship a subclass of this generic concept. As contexts, Social Relationships can have a number of Parameters, Roles and single Course as components.

A typical Role is the Relationship Role, a subclass of the Social Role concept introduced by [Masolo et al., 2004]. An example of a relationship role is (the trivial) Friend role in a friendship relation, the Student and Professor roles in a student/advisor relationship and the Uncle/Nephew roles of kinship. Relationship Roles are restricted to be played by Natural Persons.

The course of the relationship captures the generic characteristics for the course of a certain relationship, i.e. the kinds of event and their possible sequences that characterize a certain kind of relationship. The course is related to the actual events in a particular relationship by the sequences relationship.

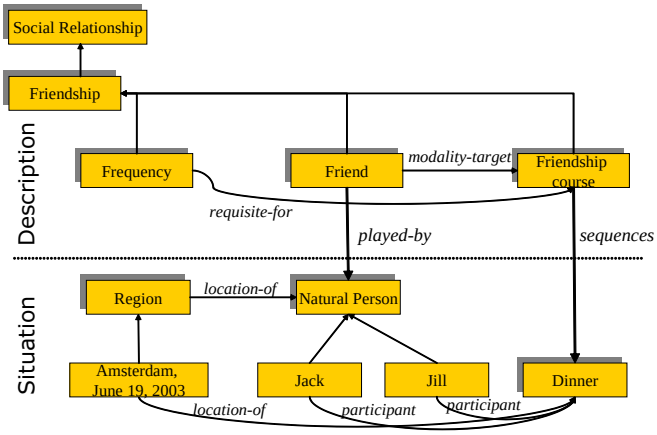


Figure 6.6: An instantiation of the Descriptions and Situations pattern for social relationships.

Characteristics of relationships such as the ones mentioned above are conceptualized as parameters, mostly a requisite for the course of the relationship. For example, frequency may be axiomatized as the average number of events in the course of the relationship within a given time unit. We recognize that softer qualities of relationships (such as the emotional content of the relationship) may be harder to capture precisely, but the engineer should strive in any case to relate it to other components of the relationship. (If the semantics cannot be captured precisely, at least the elicitation question(s) that were used to determine the quality should be documented.)

Figure 6.6 shows the representation of the friendship relation (some property instances are not shown for visualization purposes). Friendship in general is a social relationship with a single role called Friend, played by actual persons such as Jack and Jill. Friendship also has a typical course; an event such as a dinner where both Jack and Jill have participated may be related to this course, which would indicate at least that it has a significance to the development of a friendship between Jack and Jill. (Friendship is difficult to capture more precisely in this respect in that there is hardly a typical course for a friendship. Nevertheless, one may discern typical events, such as the point that the participants consider as the "beginning" of their friendship.)

In summary, we propose an advanced representation of social relationships as higher order concepts (patterns) that map to a concrete state-of-affairs representing observations about the environment. This view of relationships fits very well with the increasing availability of fine-grained electronic data (such as emails) about the history of social interactions between individuals. We expect this trend to continue in the future as we will be surrounded with more and more mobile and ubiquitous devices that collect data on our whereabouts and interactions with other people. While this may sound like a doomsday scenario, technology for tracking and observing social relationships based on

sensor data is accepted in cases where it occurs with the agreement of the participants and where it serves a well-defined purpose. For example, Matsuo et al. describe a ubiquitous system for supporting the networking of conference participants [Matsuo et al., 2006]. Through infrared sensors and later, RFID cards the system observes social networks based on the participants physical presence in the conference space and through their explicit actions: conference participants may touch their badges to readers to indicate that they have met and the system reacts by displaying information related to their meeting (e.g. shared interests). The technology has been already applied for three consecutive years at the yearly meetings of AI researchers in Japan (JSAI 2003-2005) and at the UbiComp conference in 2005.

6.4 Aggregating and reasoning with social network data

Supposing that we have started out with some data sets in traditional formats (relational databases, Excel sheets, XML files etc.) our first step is to convert them into an RDF-based syntax, which allows to store the data in an ontology store and manipulate it with ontology-based tools. In this process we need to assign identifiers to resources (an issue that we deal with in Section 6.4.1) and re-represent our data in terms of a shared ontology such as FOAF.

In case our data sets come from external sources it is often more natural to preserve their original schema. For example, in case of converting data from a relational database or Excel sheet it is natural to preserve the schema of the database or spreadsheet as represented by the table definitions or table headings. We can then apply *ontology mapping* to unify our data on the schema level by mapping classes (types) and properties from different schemas to a shared ontology such as FOAF. In effect, ontology mapping allows us to treat the data set as if it had a single shared schema. Note that research on automated methods for ontology mapping is an active research area within the Semantic Web community. It is not our primary concern as the number of ontologies involved and their size does not make it necessary to automate ontology mapping in our typical case.

The task of aggregation, however, is not complete yet: we need to find identical resources across the data sets. This is a two step process. First, it requires capturing the domain-specific knowledge of when to consider two instances to be the same. As we will see the FOAF ontology itself also prescribes ways to infer the equality of two instances, for example based on their email address. However, beyond these properties it is likely that we need to introduce some domain-specific criteria based on domain-specific properties. In order to do this, we need to consider the general meaning of equality in RDF/OWL (Section 6.4.2). In practice, only a limited part of the knowledge regarding instance equality can be captured in RDF or OWL. For example, determining equality is often considered as applying a threshold value to a similarity measure, which is a weighted combination of the similarity of certain properties. (The threshold is determined by experimentation or through machine learning.) In this and many other practical cases that involve computation we need a procedural representation of knowledge.

Once we determined the rules or procedures that determine equality in our domain, we need to carry out the actual instance unification or *smushing* (see Section 6.4.4).

Unlike much of the related work on instance unification, we consider smushing as a reasoning task, where we iteratively execute the rules or procedures that determine equality until no more equivalent instances can be found. The advantage of this approach (compared to a one-step computation of a similarity measure) is that we can take into account the learned equalities in subsequent rounds of reasoning.

We will discuss the advantages and disadvantages of using rule-based reasoners and Description Logic reasoners for this task and the trade-off between forward- and backward chaining reasoning. We will also outline the approach in case we need to combine procedural and rule based reasoning.

6.4.1 Representing identity

One of the main advantages of RDF over other representation formalisms such as UML is the possibility to uniquely identify resources (instances, classes, as well as properties). The primary mechanism for this is the assignment of URIs to resources. Every resource, except blank nodes is identified by a URI.

However, in practice it is often the case that there are a number of candidates for identifiers. For example, in the publishing world a number of identifier schemes are in use. Standard schemes such as ISBN and ISSN numbers for books and periodicals as well as DOIs (Digital Object Identifiers) are widely adopted, while each publisher and online repository also maintains its own identifier scheme. Further, publications that are accessible online can be represented by their URL. All of these identifiers are unique (an identifier identifies a single resource), but mostly not single (a resource may be identified by multiple identifiers). This is the case with URLs (the same publication may be posted at multiple URLs), but it happens even with DOIs whose registration is centralized that the same publication is assigned two or more identifiers.

Multiple identifiers can be represented in RDF in two separate ways. First, one can introduce a separate resource and use the identifiers as URIs for these resources. Once separate resources are introduced for the same object, the equality of these resources can be expressed using the *owl*:*sameAs* property (see the following section). The other alternative is to chose one of the identifiers and use it as a URI.

Note that in all cases resource identifiers need to conform to the URI specification and good practice. Many modern identifier schemes such as DOIs have been designed to conform to the URI specification. Other identifier schemes can be recoded as URIs with the new[15] info: prefix (protocol), which is regulated in [de Sompel et al., 2006]. It is also a common practice to create URIs within a web domain owned or controlled by the creator of the metadata description. For example, if the registered domain name of a public organization is `http://www.example.org` then a resource with the identifier 123 could be represented as `http://www.example.org/id/123`. This satisfies the guidelines for good URIs, in particular that good URIs should be unique and stable.[16] The first criterion means that good URIs should be unambiguous or at least should be chosen such that it is unlikely that someone else would use the same URI for something

[15] April 2006

[16] See `http://www.w3.org/Addressing/` and `http://esw.w3.org/topic/GoodURIs`

different. This is important because resources with the same URI but different intended meanings are likely to result in inconsistencies (a *URI clash*). The second criterium is also important because there is no way to rename resources (to reassign URIs). Once a URI changes the only solution is to introduce a new resource and assert its equivalence with the old resource. However, in large scale systems such as the web there is no way to notify remote systems of the new identifier and they are likely to continue referencing the resource by its old identifier. Thus it is bad idea for example to encode unstable property values in identifiers as done in [Portwin and Parvatikar, 2006], where publications are assigned identifiers based on e.g. the journal, the volume and the page number. If there turns out to be a data entry mistake in one of the values, the identifier becomes obsolete and is potentially ambiguous.

The use of http:// URIs has been somewhat complicated by the long outstanding issue of whether such URIs could be used for only web resources or also for abstract concepts (which has been the practice), and if yes, what should web servers respond to HTTP requests related to such URIs.[17] The resolution of this issue is that the response code of the web server should indicate whether the URI denotes a web resource (HTML page, image about a person, publication etc.) or some abstract concept (such as a person, a publication etc.) The response code used by web servers to indicate success should be used only for web resources.

A practical consequence is that for abstract concepts one should not choose URIs that are recognized by a server, e.g. the location of an existing HTML page, as this would be a case of URI clash. While the decision also opens the road to look up metadata about abstracts concepts using the HTTP protocol there are currently very few web servers configured to support this.

6.4.2 On the notion of equality

RDF and OWL (starting with OWL Lite) allow us to identify resources and to represent their (in)equality using the *owl*:*sameAs* and *owl*:*differentFrom* properties. However, these are only the basic building blocks for defining equality. In particular, the meaning of equality with respect to the objects of the domain depends on the domain itself and the goals of the modelling. For example, it is part of the domain knowledge what it takes for two persons or publications to be considered the same, i.e. what are the characteristics that we take into account. Further, depending on the level of modelling we may distinguish, for example, individual persons or instead consider groups of persons (e.g. based on roles) and consider each individual within the group to be equivalent (role equivalence).

Nevertheless, it is interesting to take a short de-tour to the domain of philosophy, where a number of attempts have been made over the centuries to capture the general characteristics of the notions of identity and equality (indiscernibility). The most well-known formulation of equality was given by Wilhelm Gottfried Leibniz in his *Discourse on Metaphysics*. The Identity of Indiscernibles or Leibniz-law can be loosely phrased

[17] The issue has been codenamed httprange-14 by the Technical Architecture Group (TAG) of the W3C and has been on the agenda for four years, see `http://www.w3.org/2001/tag/issues.html#httpRange-14`

as the principle of all things being identical unless we are able to distinguish (discern) them or put it differently: no two distinct objects have exactly the same properties.[18] The Leibniz-law is formalized using the logical formula given in Formula 6.1. The converse of the Leibniz-law is called Indiscernibility of Identicals and written as Formula 6.2. The two laws taken together (often also called Leibniz-law) are the basis of the definition of equality in a number of systems.

$$\forall P : P(x) \leftrightarrow P(y) \rightarrow x = y \tag{6.1}$$

$$\forall P : P(x) \leftrightarrow P(y) \leftarrow x = y \tag{6.2}$$

The reflexive, symmetric and transitive properties of equality follow from these definitions. Notice that both formulas are second-degree due to the quantification on properties. This quantification is also interesting because it provides the Leibniz-law different interpretations in open and closed worlds. Namely, in an open world the number of properties is unknown and thus the Leibniz-law is not useful in practice: we can never conclude that two resources are equal since we can never be certain whether there is another property out there that could allow us to distinguish them. In a closed world we can possibly iterate over all properties to check if two resources are equal; if we can distinguish them by some property we can assume they are equal.

In practice, a closed world assumption can be equally undesirable as an open one. In most cases we have almost complete information about our resources, but still we may not want two resources to be considered identical just because of a lack of information. For example, if we have two resources and we only know that they are the same gender, we may not want to assume they are identical (which would be the consequence in a closed world).

Philosophical ontologists have also argued against the Leibniz-law in the original form because it is stronger than our natural notion of equality. Consider for example a perfectly symmetrical space with two perfect spheres at some distance d to each other. Our natural intuition would consider the two spheres indistinguishable. However, they can be distinguished as the first sphere is distance d from the second sphere, which is not true for the second sphere (it is zero distance to itself). The solution to this problem is to limit the kind of properties to be considered, in particular to exclude impure, extrinsic properties such as distance to other instances.

The same strategy can be followed in closed worlds to introduce weaker notions of equality at will. For example, one might specify the set of properties to be checked and exclude such properties as *foaf*:*based_near*, which gives the geo-location of the individual or transient properties such as *foaf*:*gender*, which may change during the lifetime of an individual.

Lastly, let's consider the relationship of the Leibniz-law to OWL and the semantics of the *owl*:*sameAs* relationship. First, we should note that the properties we are interested in are the statements that are made about a resource. We could always distinguish two resources for example by their URIs. However, we do not want to consider the URI a

[18] `http://plato.stanford.edu/entries/identity-indiscernible/`

property of the resource, since this would be too strong of a notion of equality. (Resources could never be equal, only bNodes.)

The semantics of OWL is built on an open world assumption, which means that the Leibniz-law cannot be used to infer identity, not even if we reduce the property space even further. However, we can still infer the equality of instances by necessity (see the following Section).

On the other hand, the semantics of *owl*:*sameAs* conforms to Formula 6.2. Namely, *owl*:*sameAs* restricts the interpretation of the theory to those models where the two symbols denote the same object and thus they must be indiscernible in the sense that they are interchangeable in statements:

$$\begin{aligned}
&(s_1, \textit{owl:sameAs}, s_2) \wedge (s_1, p, o) \rightarrow (s_2, p, o) \\
&(p_1, \textit{owl:sameAs}, p_2) \wedge (s, p_1, o) \rightarrow (s, p_2, o) \\
&(o_1, \textit{owl:sameAs}, o_2) \wedge (s, p, o_1) \rightarrow (s, p, o_2)
\end{aligned} \tag{6.3}$$

The reflexive, symmetric and transitive properties of sameAs also follow:

$$\begin{aligned}
&\forall s : (s, \textit{owl:sameAs}, s) \\
&(s_1, \textit{owl:sameAs}, s_2) \rightarrow (s_2, \textit{owl:sameAs}, s_1) \\
&(s_1, \textit{owl:sameAs}, s_2) \wedge (s_2, \textit{owl:sameAs}, s_3) \rightarrow (s_1, \textit{owl:sameAs}, s_3)
\end{aligned} \tag{6.4}$$

Note that it is not inconsistent to have resources that are *owl*:*sameAs* but have different stated properties, e.g. Formula 6.5 is not an inconsistent ontology. The explanation lies again in the open world assumption: we can assume that the missing statements (that s_1 has the *foaf*:*name* Paul and s_2 has the *foaf*:*name* John) exist somewhere. In a closed world this ontology would be inconsistent.

$$\begin{aligned}
&(s_1, \textit{owl:sameAs}, s_2) \\
&(s_1, \textit{foaf:name}, \text{"John"}) \\
&(s_2, \textit{foaf:name}, \text{"Paul"})
\end{aligned} \tag{6.5}$$

6.4.3 Determining equality

In our case, we are interested in capturing knowledge about the identity of resources that can lead to conclude the (in)equality of resources.

In OWL there are a limited set of constructs that can lead to (in)equality statements. Functional and inverse functional properties (IFPs) and maximum cardinality restrictions in general can lead to conclude that two symbols must denote the same resource when otherwise the cardinality restriction could not be fulfilled. For example, the *foaf*:*mbox* property denoting the email address of a person is inverse-functional as a mailbox can only belong to a single person. As another example, consider a hypothetical *ex*:*hasParent* property, which has a maximum cardinality of two. If we state that a single person has

three parents (which we are allowed to state) an OWL reasoner should conclude that at least two of them has to be the same. Once inferred, the equality of instances can be stated by using the *owl*:*sameAs* property.[19] There are more ways to conclude that two symbols do not denote the same object, which is expressed by the *owl*:*differentFrom* property. For example, instances of disjoint classes must be different from each other.

Often, the knowledge that we need to capture cannot be expressed in OWL, only in more expressive rule languages. A typical example is the case of complex keys: for example, we may want to equate two persons only if both their names and their birth dates match. This is not possible to express in OWL but can be captured in Horn logic[20], a typical target for rule languages for the Semantic Web (see e.g. [Horrocks et al., 2004]).

However, there are even situations when the expressivity of rule languages is not sufficient either. In our early work on one of the first ontology-based Knowledge Management systems we have already noted that several elementary reasoning steps could not be expressed in declarative, logic based formalisms [Mika, 2002]. Functions for simple data type manipulation such as the concatenation of literals is not part of either DL or rule languages. A an example of this is the matching of person names.

In our applications, person names are matched by separating the first and the last name in the first step. We apply a simple heuristic for determining the cutting point. (It is not possible to completely automate the separation of first names and last names in a language-independent way.[21]) Then the last names are compared in a case unsensitive manner. If they match, the first names are compared using fuzzy string matching. In both cases we have to deal with abbreviations, e.g. to match *F.v. Harmelen* against *Frank van Harmelen*.

This algorithm is easy to describe in a programming language or more powerful transformation languages such as XSLT, which contains the necessary datatypes, data manipulation functions and the three basic constructs of procedural programming (the sequence operator, the if statement and the for loop). Rule languages in a Semantic Web context, however, lack this expressive power.

The solution is to combine declarative and procedural reasoning as described in [Mika, 2002] or as proposed by Korotkiy [Korotkiy and Top, 2006]. Similar to procedural attachments, the main idea is to compute certain relationships with components/services that conform to a particular interface. Reasoning is a combination of calling these services and executing a regular Description Logic or rule-based reasoner.

[19] Note that the *owl*:*sameAs* property can be used to state the equivalence of classes and properties, but only in OWL Full. (This requires classes and properties to be treated as individuals.)

[20] In logic, a Horn clause is a disjunction of literals with at most one positive literal, i.e. Horn clauses are formulas of the form $p \wedge q \wedge \cdots \wedge t \rightarrow u$

[21] It is easy to see that the separation of first name and last name is a very complex natural language problem. For example, encountering the Dutch name *Mirjam Huis in 't Veld* most foreigners assume that the word *Huis* belongs to the first name. However, it belongs to the last name, which means "House in the Field". As another example consider languages such as Hungarian, where first name is written last and last name is written first, as in the name of the author of this book: *Mika Péter*.

6.4.4 Reasoning with instance equality

Reasoning is the inference of new statements (facts) that *necessarily* follow from the set of known statements. As discussed in the previous Chapter, what follows from a set of statements is determined by the semantics of the language. Every fact we add to our knowledge base has a meaning in the sense that it represents some constraint on the mapping between the symbols of our representation and some domain of interpretation. In other words, every piece of additional knowledge excludes some unintended interpretation of our knowledge base.

In most sufficiently complex languages, an infinite number of new statements could be inferred from any non-trivial knowledge base. In practice, however, we are not interested in all statements that might be deduced from a knowledge base, but only those that are relevant to some task. In other words, we would like to complete our knowledge base to contain all the important knowledge relevant to that task.

There are a number of choices one has to consider when implementing reasoning, which mostly concern trade-offs between efficiency and scalability. In the following, we review some of the options and choices to be made.

Description Logic versus rule-based reasoners

Our task is instance equality reasoning, i.e. the inference of *owl*:*sameAs* and *owl*:*differentFrom* statements. In general, OWL (DL) reasoners are of limited use in performing this kind of reasoning. As mentioned in the previous Chapter, our representation needs are also slightly different from the OWL language: we use a limited set of OWL constructs (corresponding to the unofficial OWL Tiny variant), while some of our knowledge can only be captured by rules.

Nevertheless, we can use OWL reasoners to reason with the part of the knowledge that can be expressed in OWL. As many other practical applications, however, we do not benefit from the theoretical results about the efficiency (complexity) of OWL DL reasoning, which were the original motivation behind the design of the OWL language. In particular, OWL DL was designed with the idea that the language should be as expressive as possible with the limitation that it should be also *decidable* and that it should be possible to create *efficient* reasoners for it.

In many practical cases OWL is more expressive than necessary. Decidability (the guarantee to find an answer in finite time) or completeness (the guarantee to find all the complete answers) are also not an issue in cases where the time or memory available for the reasoning is limited. Lastly, results about the complexity of OWL DL are theoretical, because they concern worst-case complexity. In practice, only the average case matters as the worst cases may be extremely rare. Further, when resources are limited we are not interested in the asymptotic behavior of the algorithm; on a short scale an approximate anytime algorithm with exponential complexity may be faster than a linear one.

Description Logic reasoners are designed to support primary tasks of classification[22] and the consistency checking of ontologies. Other reasoning tasks are usually reformu-

[22] Determining the complete subclass hierarchy of the ontology, i.e. finding implicit *rdfs*:*subClassOf* relations.

lated as consistency checking. In the case of equality reasoning this means that if we want to check whether two instances are necessarily the same, we add the opposite to the ontology and check for inconsistency. If the reasoner signals inconsistency, we can be certain that the instances are the same and add that to our knowledge base. (As discussed before the lack of inconsistency would not mean that the instances are necessarily different, we might simply lack enough information.) Unfortunately, this kind of reasoning is very inefficient in practice as we need to perform a consistency check for every pair of instances in the ontology.

A better alternative is to consider rule-based reasoning and this is the method we explored in our work in combination with procedural attachments. The semantics of RDF(S) can be completely expressed using a set of inference rules, which are given in [Hayes, 2004]. Further, a significant part of the OWL semantics can be captured using rules and this contains the part that is relevant to our task. Horst shows a partial rule-based axiomatization of OWL in [ter Horst, 2005]. These rules are implemented among others by the OWLIM reasoner[23], but can also be used with any sufficiently expressive rule-based reasoner such as the query language or the custom-rule based reasoner of Sesame[24].

Forward versus backward chaining

Rules can be used either in a forward-chaining or backward-chaining manner with different trade-offs. Forward chaining means that all consequences of the rules are computed to obtain what is called a *complete materialization* of the knowledge base. Typically this is done by repeatedly checking the prerequisites of the rules and adding their conclusions until no new statements can be inferred. The advantage of this method is that queries are fast to execute since all true statements are readily available. (The reasoning is performed before queries are answered, typically right after adding some new knowledge.) The disadvantage is that storing the complete materialization often takes exorbitant amounts of space as even the simplest RDF(S) ontologies result in a large amount of inferred statements (typically several times the size of the original repository)[25], where most of the inferred statements are not required for the task at hand. Also, the knowledge base needs to be updated if a statement is removed, since in that case all other statements that have been inferred from the removed statements also need to be removed (if they cannot be inferred in some other way).

With backward-chaining, rules are executed "backwards" and on demand, i.e. when a query needs to be answered. While with forward chaining we check the prerequisites of rules and add all their conclusions, here we are given a conclusion and check whether it is explicitly stated or whether it could inferred from some rule, either because the prerequisites of the rule are explicitly stated to be true or again because they too could be inferred from some other rule(s). The drawback of backward-chaining is longer query execution times.

[23] `http://www.ontotext.com/owlim/`

[24] `http://www.openrdf.org`

[25] In fact, the number of RDF statements that follow from any RDF(S) model is infinite: there is an infinite number of axiomatic triples due to container-membership properties [Hayes, 2004].

The advantage of a rule-based axiomatization is that the expressivity of the reasoning can be fine-tuned by removing rules that would only infer knowledge that is irrelevant to our reasoning task. For example, the general RDF(S) inference rule which says that all entities that occur as subjects of a statement are instances of the *rdfs*:*Resource* class can be ignored if it is irrelevant for our reasoning what the members are of this class. Especially in the case of forward-chaining we can save significant amounts of space by not having to store this irrelevant (and rather trivial) knowledge about every resource. (In the backward chaining case we do not gain anything if we would not ask any queries anyway about the membership of this class.)

In the Flink system (see Section 7.2), we first used the built-in inference engine of the ontology store. However, some of the rules required cannot be expressed declaratively and thus we implemented our own identity reasoner in the Java language. The reasoning is performed using a combination of forward- and backward chaining in order to balance efficiency and scalability. The rules that are used to infer the equality of instances are executed iteratively in a forward chaining manner by querying the repository for the premise(s) and adding the consequents of the rules. The semantics of the resulting *owl*:*sameAs* statements is partly inferred in a backward-chaining manner.

In particular, the rules expressed in Formulae 6.4 are added in a forward-chaining manner, while those in Formulae 6.3 are dealt with at query time by rewriting queries. For example, consider the following SeRQL query which asks for the email address of the instance Peter:

```
SELECT email FROM
{example:Peter} foaf:mbox {email}
USING NAMESPACE
foaf = <http://xmlns.com/foaf/0.1/>,
example = <http://www.example.org/>
```

This query is rewritten in a way to also return all email addresses of other instances that are *owl*:*sameAs* this instance:

```
SELECT email FROM
{example:Peter} owl:sameAs {other},
{other} foaf:mbox {email}
USING NAMESPACE
foaf = <http://xmlns.com/foaf/0.1/>,
example = <http://www.example.org/>
```

Note that this will also return the email attached to the instance *example*:*Peter*, because of the reflexivity of the *owl*:*sameAs* property. At this point we already added for all instances that they are sameAs themselves. This query rewriting is implemented in the Elmo API introduced in Chapter 7.1.2.

The timing of reasoning and the method of representation

There are three basic variations on what point the identity reasoning is performed. Also, there is the related choice of whether to represent equivalence using the *owl*:*sameAs* relationship or to directly merge the descriptions of the equivalent instances.

In the first variation, smushing is carried out while adding data into a repository. This is the method chosen by Ingenta when implementing a large-scale publication metadata repository: when a new instance is added to the repository, it is checked whether an equivalent instance already exists in the repository [Portwin and Parvatikar, 2006].

In the Ingenta case, the descriptions are merged: only one of the identifiers is kept and all information about the resource is consolidated under that identifier. This method has the advantage that the repository size is reduced, since there are no redundant instances connected to each other with *owl*:*sameAs* relations. There is a serious potential drawback, however: if it turns out later that the two instances merged are not the same (for example, because the data were erroneous, the criteria for equality changed or an identifying property of the resource changed) it is impossible to unmerge the descriptions. Using the *owl*:*sameAs* relationship it is very easy to revise the results of equality reasoning by removing the appropriate statements.

The second variation is when the reasoning is performed after the repository has been filled with data containing potential duplicates. This is the choice we take in the Flink system (see Section 7.2). In this case again there is also a choice between representing the results by merging identifiers or by *owl*:*sameAs* relationships. However, the benefits of the first are not as obvious, because removing statements is an expensive operation with most triple stores.[26]

Lastly, reasoning can be performed at query time. This is often the only choice such as when querying several repositories and aggregating data dynamically in an AJAX interface such as the with the openacademia application described in Section 7.3. In this case the solution is to query each repository for instances that match the query, perform the duplicate detection on the combined set and present only the purged list to the end-user.

6.4.5 Evaluating smushing

Smushing can be considered as either a retrieval problem or a clustering problem. In terms of retrieval, we try to achieve a maximum precision and recall of the theoretically correct set of mappings. Note that unlike in ontology mapping where there is typically a one-to-one mapping between elements of two ontologies, in the instance unification problem a single resource may be mapped to a number of other resources. Thus we can also conceptualize this task as clustering and evaluate our clusters against the ideal clustering.

Instance unification or smushing can be considered as an optimization task where we try to optimize an information retrieval or clustering-based measure. Note that instance unification is “easier” than ontology mapping where one-to-one mappings are enforced.

[26] The reason is that the triple store needs to make sure that all other statements that can be inferred *only* from the statement that is being removed are also removed.

In particular, in ontology mapping a local choice to map two instances may limit our future choices and thus we run the risk of ending up in a local minimum. In case of smushing, we only have to be aware that in cases where we have both positive and negative rules (entailing *owl*:*sameAs* and *owl*:*differentFrom*) there is a possibility that we end in an inconsistent state (where two resources are the same as well as different). This would point to the inconsistency of our rules, e.g. that we did not consider certain kind of input data.

While there are a number of measures in the literature for measuring the success of retrieval or clustering, it depends ultimately on our application what are the costs of certain types of mistakes (false positives and false negatives) and how that measures up against the costs of reducing them.

6.5 Discussion

As shown in the previous Chapter, ontology languages offer a more flexible, distributed way of representing knowledge, which is particularly appealing in scenarios where knowledge sources under independent control need to be combined. This is the typical case of the Web, but occurs in much smaller scale scenarios all the way to integrating application data on a single computer.

In this Chapter, we have seen how to use ontology-based Knowledge Representation for modelling social individuals and social relationships. We have also noted that unlike more traditional representation mechanisms, ontology languages directly support the task of data integration on both the schema and instance levels by providing the necessary constructs to represent mappings. On the schema level, this task is called ontology mapping, which is a vibrant area of research within the Semantic Web community. Our main concern, however, has been integration on the instance level, which is also called smushing. The OWL ontology language supports this task by rigorous identification, declarative means to describe what it requires for two instances to be the same or different (e.g. using cardinality restrictions or disjointness) and to represent (in)equality using the *owl*:*sameAs* and *owl*:*differentFrom* properties.

However, a great deal of the knowledge we wanted to capture could not be expressed in the OWL language. We discuss these limitations in the next Section.

6.5.1 Advanced representations

As we have seen, some of the knowledge we wanted to express could not be captured in OWL DL. Some of the missing features that do not significantly change the complexity of the language such as the conspicuously absent *owl*:*ReflexiveProperty* are likely to appear in future revisions of OWL.[27] However, OWL will remain limited by the original complexity concerns, most of which do not apply to practical scenarios as explained in Section 6.4.4. More expressive power in representation is expected from the Rule

[27] At the time of writing there is already a proposal for OWL 1.1, see `http://www-db.research.bell-labs.com/user/pfps/owl/overview.html`.

Interchange Format (RIF) currently under development by the W3C. Although its development is primarily driven by requirements from use cases, RIF will most likely include first-order logic (FOL).

Additional expressive power can be brought to bear by using logics that go beyond predicate calculus. For example, *temporal logic* extends assertions with a temporal dimension: using temporal logic we can express that a statement holds true at a certain time or for a time interval. Temporal logic is required to formalize ontology versioning, which is also called change management [Klein, 2004]. With respect to our information integration task, changes may effect either the properties of the instances or the description of the classes, including the rules of equivalence, i.e. what it means for two instances of a class to be equivalent. In most practical cases, however, the conceptualization of classes in the ontology should be more stable than the data and the rules of equivalence should not include dynamic properties. Nevertheless, it is sometimes still desirable to match on such properties. For example, when merging person descriptions we might consider the color of a person's hair. Such a property can change its value over time, which means we have to consider the time when the assertions were true[28].

Extending logic with *probabilities* is also a natural step in representing our problem more accurately. As we have already discussed, the matching of properties often result in probabilistic statements. For example, when we match person names using a string matching algorithm we get a measure of the similarity of the two strings. This can be taken on its own or combined with similarity measures for other properties to form a probability for two persons to be the same.[29] When combining similarities we can use weights to express the relevance of certain properties. Such weights can be determined ad hoc or can be learned from training data using machine learning. The resulting probability could be represented in a probabilistic logic. In our work we apply a threshold to such probabilities in order to return to binary logic where two instances are either the same or not.

[28] Or will be true: temporal logic can be used of course to represent statements about the future true

[29] This is different from fuzziness: in fuzzy logic the equivalence of the two instances would be expressed on a scale of [0,1]. In our case, two instances are still either the same or not the same. It is merely that we have some uncertainty of which one is true.

Chapter 7

Implementation of the methods

Developing Semantic Web applications is not significantly different from developing data-driven applications with more traditional technologies. The difference lies in the kind of data dealt with (data with formal semantics) and the emphasis on the distributed nature of applications. These are the characteristics that define a Semantic Web application, as captured for example in the definition used by the Semantic Web Challenge, a yearly competition of Semantic Web applications.[1] In order to be considered a Semantic Web application with respect to the Challenge, an application has to meet the following criteria:

1. The meaning of data has to play a central role.
    - Meaning must be represented using formal descriptions.
    - Data must be manipulated/processed in interesting ways to derive useful information and
    - this semantic information processing has to play a central role in achieving goals that alternative technologies cannot do as well, or at all;
2. The information sources used
    - should be under diverse ownership or control
    - should be heterogeneous (syntactically, structurally, and semantically), and
    - should contain substantial quantities of real world data (i.e. not toy examples).
3. It is required that all applications assume an open world, i.e. that the information is never complete.

Note that most Semantic Web applications have a web interface and thus considered by all as web applications. However, this is not a requirement: we also consider

[1] See `http://challenge.semanticweb.org`

Semantic Web applications that have a rich client interface (desktop applications) and otherwise match the criteria of using semantic technologies in a distributed environment. In other words, Semantic Web technology has an impact on the middle and data layers of applications but rarely constrains the interface in any way.

The challenges that one has to face when developing Semantic Web applications also have to do with the unusual setting of distributed, heterogeneous information sources and the need to deal with a new kind of data, namely information with formal semantics (knowledge). In a typical Web application[2] the data come from a single source (typically a relational database), which is solely under the control of the application. The meaning of the data is also known in advance. All possible reasoning with the data can be thus hard-coded in the application.

By contrast, in building Semantic Web applications one has to deal with the effects of distributed data sources and the need to integrate the kind of reasoning into the application that we introduced in the previous Chapter. However, even before dealing with such substantial issues, those plunging into Semantic Web development also need to be aware of the slate of technologies related to ontology-based knowledge representation such as standard ontology languages (RDF/OWL), representations (RDF/XML), query languages (SPARQL) etc.

All of these result in the fact that Semantic Web application development requires a very steep learning curve, which has long hindered the adoption of Semantic Web technology. Many developers left dissatisfied especially when applying these "heavyweight" technologies to trivial problems, where the returns are minimal. In fact, the advantages of semantic-based application development often manifest themselves on the long run as applications need to be extended or data sources need to be re-purposed. Lastly, there is also the chicken and egg problem of lacking applications due to a lack of data in RDF format and lacking data due to a lack of applications.

In order to lower the threshold for developing Semantic Web applications in general and social network applications in particular, we have developed two key components that allow developers to create applications at higher levels of abstraction. Our library named Elmo, an extension of the popular Sesame triple store, enables developers to create applications at the level of domain objects (persons, publications etc.) instead of working at level of the ontology language (resources, literals and statements). Acting as a middle layer between the RDF triple store and the application Elmo hides much of the complexity from the developer. In addition, Elmo contains a number of tools for dealing with RDF data.

Another simple utility named GraphUtil maps the FOAF representation of network data to the graph model of the JUNG (Java Universal Network Graph) API, allowing the developer to deal with network data in FOAF at the level of network concepts such vertices, edges etc. and to compute network statistics such as various measures of centrality.

In the following, we briefly introduce Sesame, Elmo, the GraphUtil utility, and then show two applications using these core components. These typical Semantic Web applications with social networking aspects highlight the features of our tools and the advan-

[2] We can think of the prototypical online book store.

tages of semantic technology in building intelligent applications on top of heterogeneous, distributed information sources.

7.1 Developing Semantic Web applications with social network features

In the following we introduce first Sesame, a general database for the storing and querying RDF data. Sesame has been developed by Aduna (formerly Aidministrator), but available as open source (currently under LGPL license). Next, we describe the Elmo API, a general purpose extension to Sesame with specific tools for collecting and aggregating RDF data from distributed, heterogeneous information sources. Elmo has been developed by the author and is available under the same conditions as Sesame.

Lastly, we introduce a simple utility called GraphUtil which facilitates reading FOAF data into the graph object model of the Java Universal Network Graph (JUNG) API. GraphUtil is open source and available as part of Flink (see Section 7.2).

7.1.1 Sesame

Sesame is one of the most popular RDF triple stores.[3] A triple store is much like a database for RDF data: it allows creating repositories, storing RDF in a repository and querying the data using any of the supported query languages including Sesame's own SeRQL language and SPARQL.[4]

While there is no update operation, individual statements can be added and removed from the repository. RDF data can be added or extracted in any of the supported RDF representations including the RDF/XML and Turtle languages introduced in Chapter 5. Sesame can store the RDF data in a variety of back-ends: in memory, on the disk or using a relational database.

As most RDF repositories, Sesame is not only a data store but also integrates reasoning. Sesame has a built-in inferencer for applying the RDF(S) inference rules (see Section 5.6). While Sesame does not support OWL semantics, it does have a rule language that allows to capture most of the semantics of OWL, including the notion of inverse-functional properties and the semantics of the *owl*:*sameAs* relationship (see Section 6.4.2).

Sesame provides a Java client library for developers which exposes all the above mentioned functionality of a Sesame repository using method calls on a Java object called *SesameRepository*. Queries, for example, can be executed by calling the *evaluateTableQuery* method of this class, passing on the query itself and the identifier of the query language. The result is another object (*QueryResultsTable*) which contains the result set in the form of a table much like the one shown in the web interface (see Figures 7.1 and 7.2). Every row is a result and every column contains the value for a given

[3] Elmo and Sesame are hosted at `http://www.openrdf.org`

[4] While SPARQL has the advantage in terms of standardization, it is also minimal by design. SeRQL is a more expressive query language.

variable. The values in the table are objects of type *URI*, *BNode* or *Literal*, the object representations of the same notions in RDF. For example, one may call the *getValue*, *getDatatype* and *getLanguage* methods of *Literal* to get the String representation of the literal, its datatype and its language.

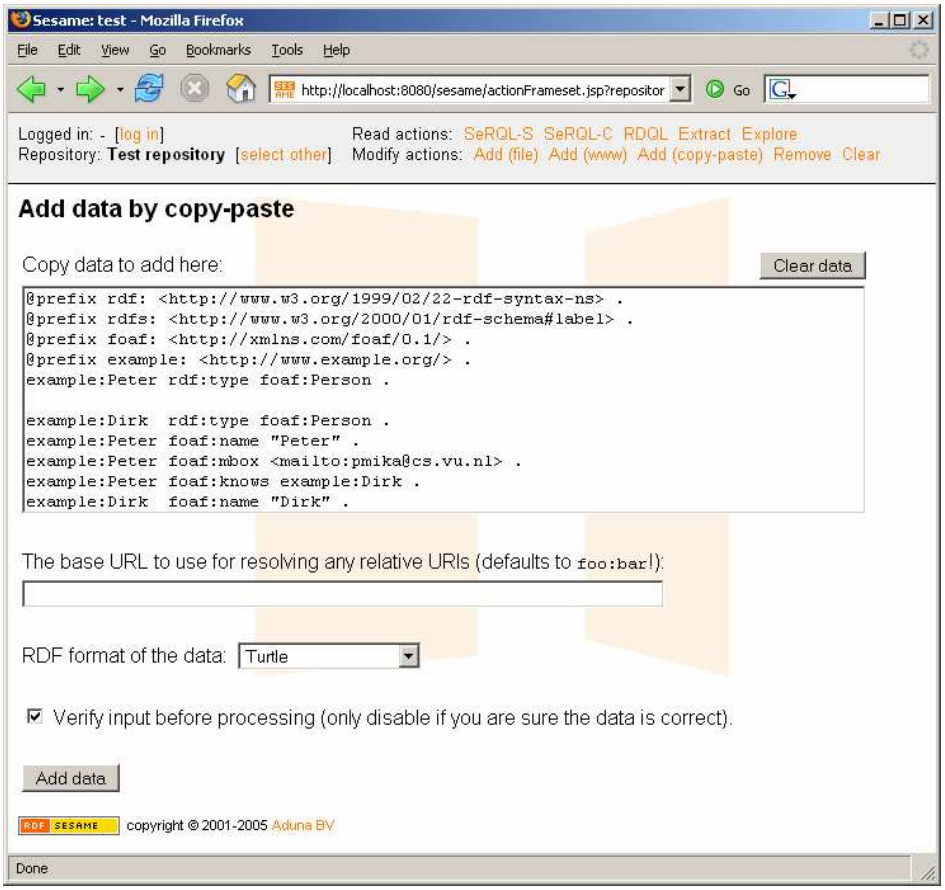


Figure 7.1: Adding data to a Sesame repository using the web interface. Here we add the data used in the examples of Chapter 5.

7.1.2 Elmo

While Sesame's client library is appropriate for manipulating RDF data at a very low level, when developing Semantic Web applications it is more suitable to treat RDF data at an ontological level. The Elmo API allows developers to create applications that work with RDF/OWL repositories at the level of domain ontologies, instead of working on the level of the RDF/OWL languages.

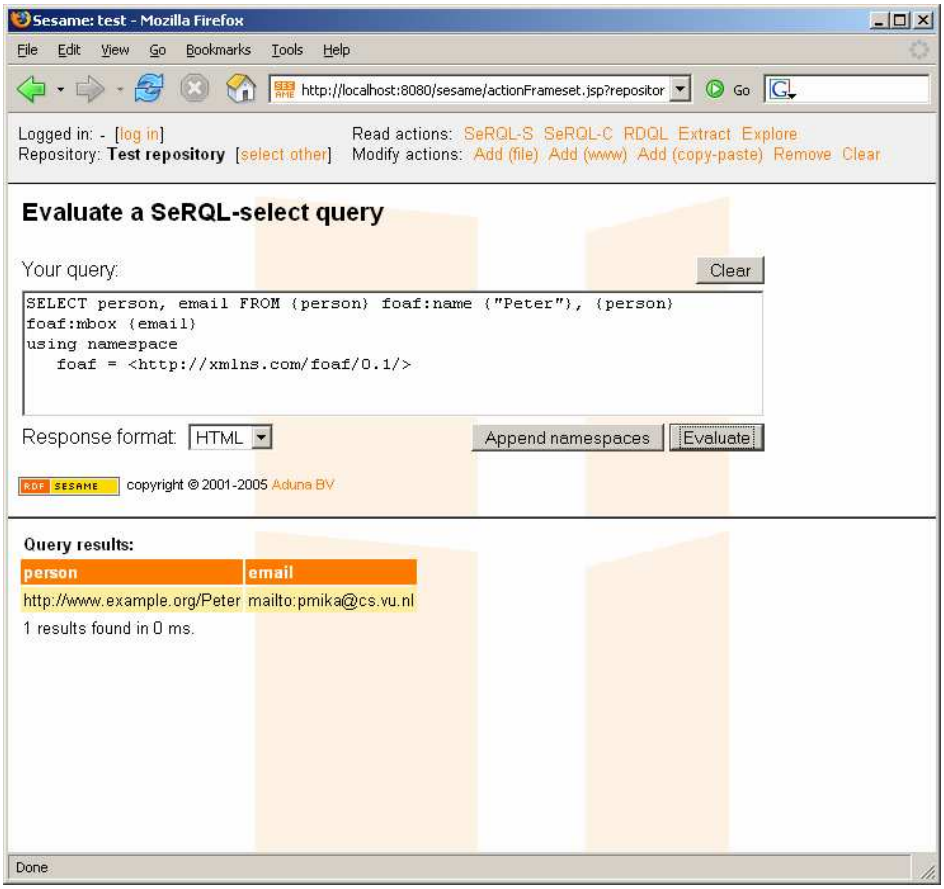


Figure 7.2: Querying data through the web interface of Sesame. The results are displayed below in a table.

In particular, Elmo provides static object models for the most popular Web ontologies, including FOAF, RSS 1.0 and Dublin Core. This means that every domain class has an equivalent JavaBeans representation with the corresponding properties, for example there is *Person* JavaBean with the properties of *foaf*:*Person*. Getting and setting these properties manipulates the underlying RDF data.[5]

This higher level of representation significantly simplifies development. For example, a simple FOAF profile can be created in ten lines of Java code (see Figure 7.3).

[5]Static object models for other ontologies can be created manually. In future versions of Elmo, it will be possible to generate them automatically from the ontology using existing code generators such as Jastor, see `http://jastor.sourceforge.net/`. Existing JavaBean implementations can be also reused by annotating them (a feature of Java 5) or by making them implement the DynaBean interface from the Apache Beanutils package, see `http://jakarta.apache.org/commons/beanutils/`.

```
ElmoSession session = new ElmoSession();
URI jackURI = session.createURI("http://www.example.org/#jack");

Person jack = (Person) session.getInstance(jackURI, Person.class);

jack.setName(session.createLiteral("Jack"));
jack.setMbox(session.createURI("mailto:jack@example.org"));
jack.setMbox(session.createURI("mailto:jack@work.example.org"));

RdfDocumentWriter writer = new AbbreviatedRdfXmlWriter(System.out);
writer.setNamespace("foaf", Person.FOAF_NS);
writer.startDocument();
new PersonWriter().writeRDF(writer, jack);
writer.endDocument();
```

Figure 7.3: Creating and writing out a FOAF profile in Elmo.

As we see in this example, all the interaction with the underlying repository is encapsulated by the *ElmoSession* class, which is used to create the JavaBean (or retrieve it, in case it already existed). After setting some of the properties of the *Person* instance, we write it out as an RDF/XML document.

The *ElmoSession* class is also used to provide higher level functionality such as caching. Information read from the repository is cached for further queries. (Similarly, writes are also cached until the transaction is committed. The default, however, is automatic commit.) Caching also involves predicting the kind of queries the user is likely to ask and pre-loading the information accordingly. Already when a resource is first accessed all the properties of that resource are preloaded. Another strategy requires keeping track of the queries from which resources have been retrieved. If later a property is read on such a resource, the same property is retrieved for all the resources originating from the same query.

Optionally, the query re-writing introduced in Section 6.4.4 can also be performed by Elmo: for example, when executing the *getName* method on a Person instance not only the names of current instance is returned, but also all the names of all instances that are *owl*:*sameAs* the current instance.

Another important functionality implemented on top of Sesame is the storing and retrieving of context information. In distributed scenarios we often want to store information about the provenance of statements. For example, in the case of collecting FOAF profiles from the Web we might want to keep track of where the information came from (the web address of the profile) and the time we last looked at it. However, context information is also important for centralized sites where users post information and we want to keep track of which user added a piece of information in order to establish ratings or trust. Context information is stored using RDF reification without the developer having to deal with the details of this representation.[6]

[6]Starting from Sesame 2.0, the repository natively supports the storage and querying of context information. In effect, every triple becomes a *quad*, with the last attribute identifying the context. Contexts are identified by

Elmo also contains a number of tools to work with RDF data. The Elmo *scutter* is a generic RDF crawler that follows *rdfs*:*seeAlso* links in RDF documents, which typically point to other relevant RDF sources on the web.[7] RDF(S) seeAlso links are also the mechanism used to connect FOAF profiles and thus (given a starting location) the scutter allows to collect FOAF profiles from the Web.

Several advanced features are provided to support this scenario:

- Blacklisting: sites that produce FOAF profiles in large quantities are automatically placed on a blacklist. This is to avoid collecting large amounts of uninteresting FOAF data produced by social networking and blogging services or other dynamic sources.
- Whitelisting: the crawler can be limited to a domain (defined by a URL pattern).
- Metadata: the crawler can optionally store metadata about the collected statements. This metadata currently includes provenance (what URL was the information coming from) and timestamp (time of collection)
- Filtering: incoming statements can be filtered individually. This is useful to remove unnecessary information, such as statements from unknown namespaces.
- Persistence: when the scutter is stopped, it saves its state to the disk. This allows to continue scuttering from the point where it left off. Also, when starting the scutter it tries to load back the list of visited URLs from the repository (this requires the saving of metadata to be turned on).
- Preloading from Google: the scutter queue can be preloaded by searching for FOAF files using Google
- Logging: The Scutter uses log4j to provide a detailed logging of the crawler.

The task of the Elmo *smusher* is to find equivalent instances in large sets of data, which is the problem we discussed in Section 6.4. This is a particularly common problem when processing collections of FOAF profiles as several sources on the Web may describe the same individual using different identifiers or blank nodes.

Elmo provides two kinds of smushers that implement strategies to smushing. The first kind of smusher uses class-specific comparators for comparing instances. Implementations are given for comparing *foaf*:*Person* objects based on name, email addresses and other identifying properties. There is also a comparator for comparing publications based on a combination of properties.

The second kind of smusher compares instances in a repository based on a certain property, i.e. in this case smushing proceeds property-by-property instead of instance-by-instance. For example, inferring equality based on inverse functional properties can be done with a single query for all such properties:

resources, which can be used in statements as all other resources. Contexts (named graphs) can also be directly queried using the SPARQL query language supported by this version of Sesame.

[7] The Elmo scutter is based on original code by Matt Biddulph for Jena.

```
CONSTRUCT {x} owl:sameAs {y} FROM
{prop} rdf:type {owl:InverseFunctionalProperty},
{x} prop {v}, {y} prop {v}
USING NAMESPACE
foaf = <http://xmlns.com/foaf/0.1/>,
example = <http://www.example.org/>,
owl = <http://www.w3.org/2002/07/owl#>
```

When resolving such a CONSTRUCT query first the graph pattern described after the FROM keyword is matched against the repository and for every occurrence the variables are bound to actual values.[8] With these bindings a set of new graphs is constructed by filling the variables in the pattern described in front of the FROM keyword. These graphs are merged and returned in a single RDF document. Notice that the query will also infer *owl*:*sameAs* relations where $x = y$, although only for instances that do have at least one value specified for at least one inverse functional property. This can be prevented by adding an additional WHERE clause.

The smushers report the results (the matching instances) by calling methods on registered listeners. We provide several implementations of the listener interface, for example to write out the results in HTML, or to represent matches using the *owl*:*sameAs* relationship and upload such statements to a Sesame repository.

Smushers can also be used as a *wrapper*. The difference between a wrapper and a smusher is that a smusher finds equivalent instances in a single repository, while a wrapper compares instances in a source repository to instances in a target repository. If a match is found, the results are lifted (copied) from the source repository to the target repository. This component is typically useful when importing information into a specific repository about a certain set of instances from a much larger, general store.

Lastly, Elmo has a framework to create *validators* that check instance data for correctness. In general, many of the RDF documents on the Web (especially documents written by hand) are either syntactically incorrect, semantically inconsistent or violate some of the assumptions about the usage of the vocabularies involved. Most of these problems result from human error. For example, many users of FOAF mistakenly assume that the value of the *foaf*:*mbox* property should be a Literal. In reality, the ontology expects a URI that encodes the email address using the mailto protocol, e.g. *mailto:pmika@cs.vu.nl*.

Syntax can be easily checked by syntax validators such as the online RDF validation service of the W3C.[9]. Inconsistency can be checked by OWL DL reasoners.[10] Elmo validators on the other hand are typically used to check certain assumptions about the

[8] From a practical perspective, it is also worth noting that the order of the graph expressions in the query does matter with respect to performance. Queries are evaluated from left to right by most engines and there it is reasonable to put in from the pattern that produces the least matches. In our case the first triple pattern contains only one variable (property) and that can only be bound to a small number of values (the number of inverse functional properties). The other two triple patterns contain three variables and thus match all statements in the repository. Putting them in front would result in a very inefficient query resolution.

[9] `http://www.w3.org/RDF/Validator/`

[10] Recall that RDF ontologies can not be inconsistent, except for the rare case of datatype inconsistency.

data that cannot be expressed in the ontology language, for example that the value of the *foaf*:*mbox* property has to begin with the mailto: prefix (protocol identifier). (The mistake of using a Literal would also be found by an OWL DL reasoner, because the *foaf*:*mbox* property is declared to be an *owl*:*ObjectProperty*.)

Validators can be used to create services that help data providers to check their data before submitting it to an application. We note that another common strategy to deal with invalid data is to create "forgiving parsers" that correct common user mistakes on the fly and without requiring the involvement of the user. This is the way web browsers work when tolerating many of the mistakes authors make when creating HTML pages. And while there are HTML validators that can be used to spot these mistakes, only a small number of web designers take the effort to use such validators. For similar reasons, Semantic Web applications also need to be built as robust as possible (e.g. accepting also Literals for the value of *foaf*:*mbox*) and not rely solely on user validation.

7.1.3 GraphUtil

GraphUtil is a simple utility that facilitates reading FOAF data into the graph object model of the Java Universal Network Graph (JUNG) API. GraphUtil can be configured by providing two different queries that define the nodes and edges in the RDF data. These queries thus specify how to read a graph from the data. For FOAF data, the first query is typically one that returns the *foaf*:*Person* instances in the repository, while the second one returns *foaf*:*knows* relations between them. However, any other graph structure that can be defined through queries (views on the data) can be read into a graph.

JUNG[11] is a Java library (API) that provides an object-oriented representation of different types of graphs (sparse, dense, directed, undirected, k-partite etc.) JUNG also contains implementations for the most well known graph algorithms such as Dijkstra's shortest path. Various implementations of the *Ranker* interface allow to compute various social network measures such as the different variations of centrality described in Section 3.3.3. We extended this framework with a new type of ranker called *PermanentNodeRanker* which makes it possible to store and retrieve node rankings in an RDF store.

Lastly, JUNG provides a customizable visualization framework for displaying graphs. Most importantly, the framework let's the developer choose the kind of layout algorithm to be used and allows for defining interaction with the graph visualization (clicking nodes and edges, drag-and-drop etc.) The visualization component can be used also in applets as is the case in Flink and openacademia.

7.2 Flink

Flink has been the first system that exploits semantic technologies for the purposes of network analysis based on heterogeneous knowledge sources and has been the winner of the Semantic Web Challenge of 2004. Flink, developed by the present author, is a general

[11] `http://jung.sourceforge.net`

system that can be instantiated for any community for which substantial electronic data is available.

The current, public instantiation of Flink[12] is a presentation of the professional work and social connectivity of Semantic Web researchers.[13] For the purposes of this website we have defined this community as those researchers who have submitted publications or held an organizing role at any of the past International Semantic Web Conferences (ISWC) or the Semantic Web Working Symposium of 2001. At the moment this is a community of 744 researchers from both academia and industry, covering much of the United States, Europe and to a lesser degree Japan and Australia (see Figure 2.4).

The information sources are largely the natural byproducts of the daily work of a community: HTML pages on the Web about people and events, emails and publications. From these sources Flink extracts knowledge about the social networks of the community and consolidates what is learned using a common semantic representation, namely the FOAF ontology.

The *raison d'ětre* of Flink can be summarized in three points. First, Flink is a demonstration of the latest Semantic Web technology. In this respect, Flink is interesting to all those who are planning to develop systems using Semantic Web technology for similar or different purposes. Second, Flink is intended as a portal for anyone who is interested to learn about the work of the Semantic Web community, as represented by the profiles, emails, publications and statistics. Hopefully Flink will also contribute to bootstrapping the nascent FOAF-web by allowing the export of the knowledge in FOAF format. This can be taken by the researchers as a starting point in setting up their own profiles, thereby contributing to the portal as well. Lastly, but perhaps most importantly, the data collected by Flink is used for the purposes of social network analysis, in particular learning about the nature of power and innovativeness in scientific communities (see Chapter 9).

7.2.1 The features of Flink

Flink takes a network perspective on the Semantic Web community, which means that the navigation of the website is organized around the social network of researchers. Once the user has selected a starting point for the navigation, the system returns a summary page of the selected researcher, which includes profile information as well as links to other researchers that the given person might know. The immediate neighborhood of the social network (the ego-network of the researcher) is also visualized in a graphical form (see Figure 7.4).

The profile information and the social network is based on the analysis of webpages, emails, publications and self-created profiles. (See the following Section for the technical details.) The displayed information includes the name, email, homepage, image, affiliation and geographic location of the researcher, as well as his interests, participation at Semantic Web related conferences, emails sent to public mailing lists and publications written on the topic of the Semantic Web. The full text of emails and publications can be accessed by following external links. At the time of writing, the system contained

[12] `http://flink.semanticweb.org`

[13] We plan to introduce a version of Flink open to external researchers who would like to experiment with data about different communities.

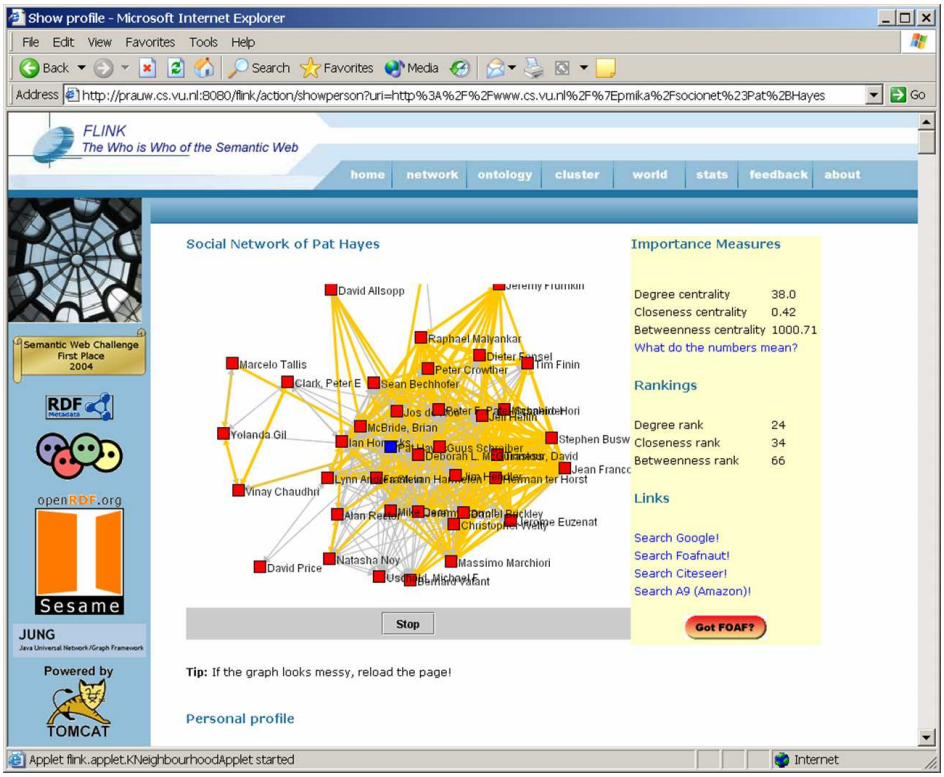


Figure 7.4: The profile of a researcher in Flink. Individual statistics (rankings) are shown on the right.

information about 7269 publications authored by members of the community and 10178 messages sent via Semantic Web-related mailing lists.

The navigation from a profile can also proceed by clicking on the names of co-authors, addressees or others listed as known by this researcher. In this case, a separate page shows a summary of the relationship between the two researchers, in particular the evidence that the system has collected about the existence of this relationship. This includes the weight of the link, the physical distance, friends, interests and depictions in common as well as emails sent between the researchers and publications written together. The information about the interests of researchers is also used to generate a lightweight ontology of the Semantic Web community. The concepts of this ontology are research topics, while the associations between the topics are based on the number of researchers who have an interest in the given pair of topics (see Figure 7.5).

An interesting feature of this ontology is that the associations created are specific to the community of researchers whose names are used in the experiment. This means that unlike similar lightweight ontologies created from a statistical analysis of generic web content, this ontology reflects the specific conceptualizations of the community that was

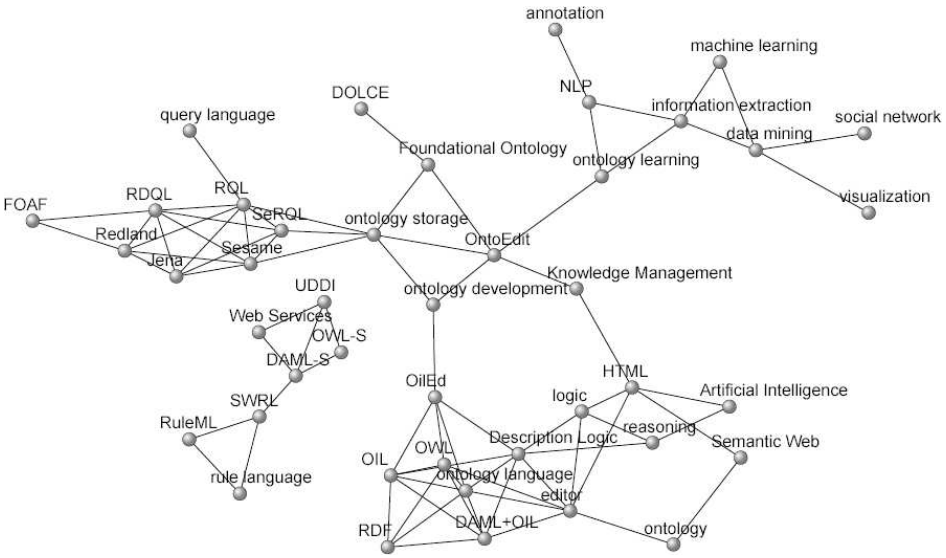


Figure 7.5: A visualization of the associations between research topics within the Semantic Web community.

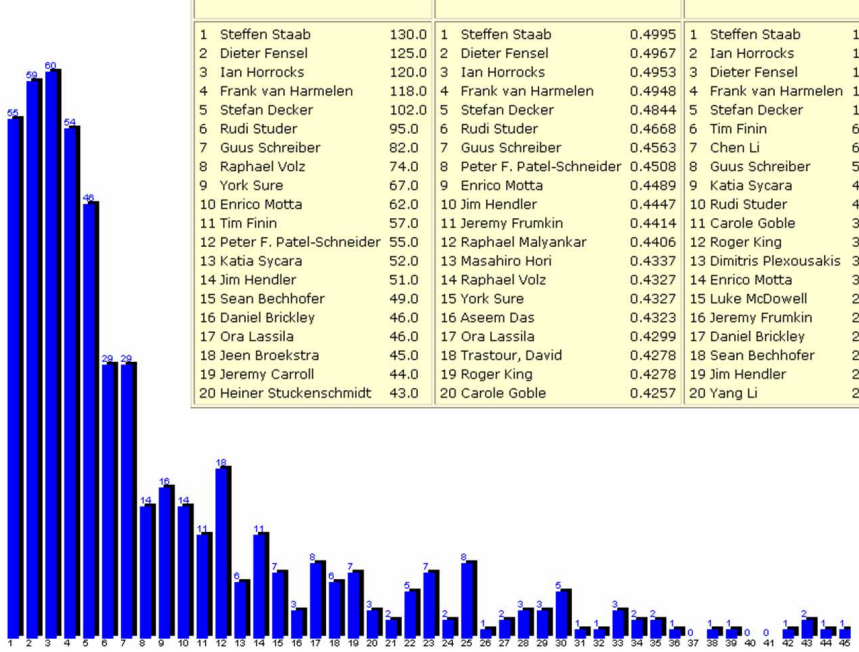
used in the extraction process (see the following Section). Also, the ontology naturally evolves as the relationships between research topics changes (e.g. as certain fields of research move closer to each other). For a further discussion on the relation between sociability and semantics, we refer the reader to Chapter 10.

The visitor of the website can also view some basic statistics of the social network. Degree, closeness and betweenness are common measures of importance or influence in Social Network Analysis, while the degree distribution attests to a general characteristic of the network itself (see Figure 7.6).

Geographic visualizations of the Semantic Web offer another overview of the network by showing the places where researchers are located and the connections between them (see Figure 2.4).

7.2.2 System design

Similarly to the design of most data-driven applications, the architecture of Flink can be divided in three layers concerned with metadata acquisition, storage and presentation, respectively. Figure 7.7 shows an overview of the system architecture with the three layers arranged from top to bottom. In the following, we describe the layers in the same order.



Indegree centrality		Closeness Centrality		Betweenness Centrality	
1 Steffen Staab	130.0	1 Steffen Staab	0.4995	1 Steffen Staab	16864.4403
2 Dieter Fensel	125.0	2 Dieter Fensel	0.4967	2 Ian Horrocks	14098.6831
3 Ian Horrocks	120.0	3 Ian Horrocks	0.4953	3 Dieter Fensel	13544.063
4 Frank van Harmelen	118.0	4 Frank van Harmelen	0.4948	4 Frank van Harmelen	12537.5246
5 Stefan Decker	102.0	5 Stefan Decker	0.4844	5 Stefan Decker	10238.5444
6 Rudi Studer	95.0	6 Rudi Studer	0.4668	6 Tim Finin	6547.5342
7 Guus Schreiber	82.0	7 Guus Schreiber	0.4563	7 Chen Li	6506.0598
8 Raphael Volz	74.0	8 Peter F. Patel-Schneider	0.4508	8 Guus Schreiber	5372.3001
9 York Sure	67.0	9 Enrico Motta	0.4489	9 Katia Sycara	4826.9771
10 Enrico Motta	62.0	10 Jim Hendler	0.4447	10 Rudi Studer	4178.6547
11 Tim Finin	57.0	11 Jeremy Frumkin	0.4414	11 Carole Goble	3882.1959
12 Peter F. Patel-Schneider	55.0	12 Raphael Malyankar	0.4406	12 Roger King	3686.1388
13 Katia Sycara	52.0	13 Masahiro Hori	0.4337	13 Dimitris Plexousakis	3337.4544
14 Jim Hendler	51.0	14 Raphael Volz	0.4327	14 Enrico Motta	3107.1791
15 Sean Bechhofer	49.0	15 York Sure	0.4327	15 Luke McDowell	2950.3047
16 Daniel Brickley	46.0	16 Aseem Das	0.4323	16 Jeremy Frumkin	2792.4088
17 Ora Lassila	46.0	17 Ora Lassila	0.4299	17 Daniel Brickley	2780.8619
18 Jeen Broekstra	45.0	18 Trastour, David	0.4278	18 Sean Bechhofer	2714.4584
19 Jeremy Carroll	44.0	19 Roger King	0.4278	19 Jim Hendler	2625.8347
20 Heiner Stuckenschmidt	43.0	20 Carole Goble	0.4257	20 Yang Li	2608.5416

Figure 7.6: A visualization of the degree distribution of the network. The inset shows the rankings of researchers according to various network measures.

Information collection

This layer of the system concerns the acquisition of metadata. Flink uses four different types of knowledge sources: HTML pages from the web, FOAF profiles from the Semantic Web, public collections of emails and bibliographic data. Information from the different sources is collected in different ways but all the knowledge that is learned is represented according to the same ontology (see the following Section). This ontology includes FOAF and minimal extensions required to represent additional information.

The web mining component of Flink employs a co-occurrence analysis technique described in Chapter 4. The web mining component also performs the additional task of finding topic interests, i.e. associating researchers with certain areas of research. The network ties, the interest associations and other metadata are represented in RDF using terms from the FOAF vocabulary such as *foaf*:*knows* for relationships and *foaf*:*topic_interest* for research interests. A reification-based extension of the FOAF model is necessary to represent association weights. (See Section 6.3.)

FOAF is the native format of profiles that we collect from the Web. FOAF profiles are gathered using the Elmo scutter, starting from the profile of the author. Our scutter is focused in that it only collects profiles related to one of the community members and it is also limited to potentially relevant statements, i.e. those triples where the predicate

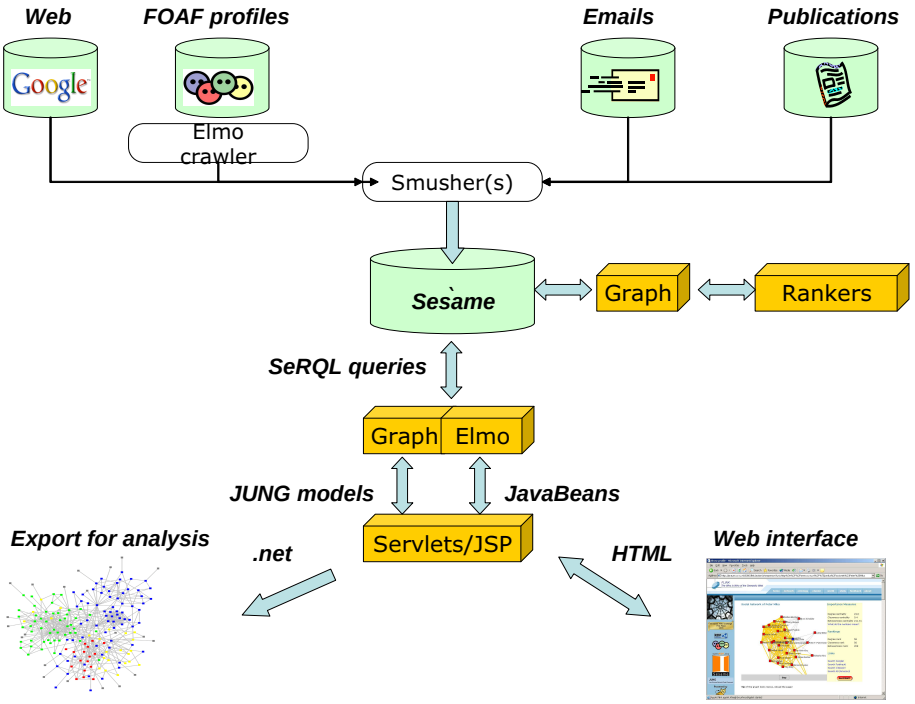


Figure 7.7: The architecture of Flink.

is in the RDF, RDF-S, FOAF or WGS-84 namespace. These restrictions are necessary to limit the amount of data collected, which can easily reach millions of triples after running the scutter for only an hour. The context features of Elmo are used to record the provenance of the statements collected. Provenance in our system consists of the source of a statement and the time it was collected.[14]

Information from emails is processed in two steps. The first step requires that the emails are downloaded from a POP3 or IMAP store and the relevant header information is captured in an RDF format, where FOAF is used for representing information about senders and receivers of emails, in particular their name (as appears in the header) and email address. (There is no common ontology for representing emails, so the rest of the header information is represented using a custom ontology.) The second step is smushing: matching the Person instances found in the email collection with the instances describing our target community. . Lastly, bibliographic information is collected in a single step by querying Google Scholar with the names of individuals (plus the disambiguation term). From the results we learn the title and URL of publications as well as the year of publication and the number of citations where available.

[14]This information is not used, only displayed at the moment. In the future it would allow to introduce features such as the ranking of information based on recency or trustworthiness.

Publication metadata is represented using the "Semantic Web Research Community" (SWRC) ontology[15] The SWRC ontology maps the types and fields of the BibTeX bibliographic description format in the most straightforward manner to RDF classes and properties. For example, the BibTeX item type InBook has the equivalent class of *swrc*:*InBook* in the ontology, with common properties such as *swrc*:*year* for the year of publication.

An alternative source of bibliographic information (used in previous versions of the system) is the Bibster peer-to-peer network [Haase et al., 2005], from which metadata can be exported directly in the SWRC ontology format.

Storage and aggregation

This is the middle layer of our system with the primary role of storing and aggregating metadata. Aggregation requires mapping the ontologies used and performing the instance reasoning described in Section 6.4.

In our case ontology mapping is a straightforward task: the schemas used are small, stable, lightweight web ontologies (SWRC and FOAF). Their mapping cause little problem: such mappings are static and can be manually inserted into the knowledge base. An example of such a mapping is the subclass relationship between the *swrc*:*Person* and *foaf*:*Person* classes or the subproperty relationship between *swrc*:*name* and *foaf*:*name*. Note that at the moment we throw away all the data we find through crawling that is not in one of the two ontologies. Incorporating knowledge in unknown schemas would require automated ontology mapping.

The aggregated collection of RDF data is stored in a Sesame server. Note that since the model is a compatible extension of FOAF, from this point the knowledge can be further processed by any FOAF-compatible tool, e.g the FOAF explorer[16]. Another example is the generic component we implemented for finding the geographical locations (latitude, longitude coordinates) of place names found in the FOAF profiles. This component invokes the ESRI Place Finder Sample Web Service[17], which provides geographic locations of over three-million place names worldwide.[18]

From a scalability perspective, we are glad to note that the Sesame server offers very high performance in storing data on the scale of millions of triples, especially using native repositories or in memory storage. Speed of upload is particularly important for the RDF crawler, which itself has a very high throughput. Unfortunately, the speed of upload drops significantly when custom rules need to be evaluated.

While the speed of uploads is important to keep up with other components that are producing data, the time required for resolving queries determines the responsiveness of the user interface. At the moment query optimization is still a significant challenge for the server. Our experience is that in many cases, the developer himself can improve the performance of a query by rewriting it manually, e.g. by reordering the terms. The

[15] See `http://ontoware.org/projects/swrc/` .

[16] `http://xml.mfd-consult.dk/foaf/explorer/`

[17] `http://www.esri.com/software/arcwebservices/`

[18] Web Service invocation is facilitated by the Apache Web Service Invocation Framework, which uses the WSDL profile of a web service to generate the code required to interact with the service.

trade-off between executing many small queries versus executing a single large query also requires the careful judgement of the developer. The trade-off is in terms of memory footprint vs. communication overhead: small, targeted queries are inefficient due to the communication and parsing involved, while large queries produce large result sets that need to be further processed on the client side.

Besides aggregation, we also use reasoning to enrich the data. For example, we infer *foaf*:*knows* relations between the senders and recipients of emails and the co-authors of publications.

Lastly, at this stage we pre-compute and store the kind of network statistics displayed in the interface. Computing these statistics takes time and therefore it cannot be done on the fly. Instead, we store the statistics in an RDF repository and read it when required. In practice, statistics are stored separately from the data because they are kept longer than the data itself: the interface also displays changes in statistics, i.e. the difference between the last two values measured for a given node using a given measure.

User interface

The user interface of Flink is a pure Java web application based on the Model-View-Controller (MVC) paradigm. The key idea behind the MVC pattern is a separation of concerns among the components responsible for the data (the model), the application logic (controller) and the web interface (view). The Apache Struts Framework used by Flink helps programmers in writing web applications that respect the MVC pattern by providing abstract application components and logic for the pattern. The role of the programmer is to extend this skeletal application with domain and task specific objects.

The model objects of Flink are Elmo beans representing persons, publications, emails etc. When requests reach the controller, all the beans that are necessary to generate the page are retrieved from the store and passed on to the view layer. The GraphUtil utility is used again to read the social network from the repository, which is also handed over to the visualization. (Much like the Elmo beans the network is also kept in memory to improve performance.)

In the view layer, servlets, JavaServer Pages (JSP) and the Java Standard Tag Library (JSTL) are used to generate a front-end that hides much of the code from the designer of the front-end. This means that the design of the web interface may be easily changed without affecting the application and vice versa.

In the current interface, Java applets are also used on parts of the site for interactive visualization of social networks. These applets communicate with a servlet that retrieves the part of the network to be visualized from the repository and sends back a serialized form to the applet, which then computes the layout. The user can pan and zoom the image as required.

We consider the flexibility of the interface to be important because there are many possibilities to present social networks to the user and the best way of presentation may depend on the size of the community as well as other factors. The possibilities range from “text only” profiles (such as in Orkut[19]) to fully graphical browsing based on net-

[19] `http://www.orkut.com`

work visualization (as in the FOAFnaut[20] browser). The uniqueness of presenting social networks is also the primary reason that we cannot benefit from using Semantic Web portal generators such as HayStack [Quan and Karger, 2004], which are primarily targeted for browsing more traditional object collections.

The user interface also provides mechanisms for exporting the data. For more advanced analysis and visualization options, the data can be downloaded in the format used by Pajek, a popular network analysis package [Batagelj and Mrvar, 1998]. Users can also download profiles for individuals in RDF/XML format. Lastly, we provide marker files for XPlanet, an application that visualizes geographic coordinates and geodesics by mapping them onto surface images of the Earth (see Figure 2.4).

7.3 openacademia

Information about scientific publications is often maintained by individual researchers. Reference management software such as EndNote and BibTeX help researchers to maintain a personal collection of bibliographic references. (These are typically references to one's own publications and also those that have been read and cited by the researcher in his own work.) Most researchers and research groups also have to maintain a web page about publications for interested peers from other institutes. Typically, personal reference management and the maintenance of web pages is a separate effort: the author of a new publication adds the reference to his own collection, updates his web page and possible that of his research group. From then on it is waiting for other researchers to discover the newly added publication.

The openacademia system removes the unnecessary duplication of effort involved in maintaining personal references and webpages. It also solves the problem of creating joined publication lists for webpages at the group or institutional level. At the same time it gives a new way of instantly notifying interested peers of new works instead of waiting for them to visit the web page of the researcher or the institute.

openacademia is a distributed system on its own. A public openacademia website is available on the Web for general use, i.e. anyone can submit his own publications to this service.[21] openacademia can also be installed at research groups locally in order to collect and manage the shared publication metadata of the group.

7.3.1 The features of openacademia

The most immediate service of openacademia is the possibility to generate an HTML representation of one's personal collection of publications and publish it on the Web. This requires filling out a single form on the openacademia website, which generates the code (one line of JavaScript!) that needs to be inserted into the body of the homepage. The code inserts the publication list in the page dynamically and thus there is no need to update the page separately if the underlying collection changes (see Figure 7.8).

[20] `http://www.foafnaut.org/`

[21] `http://www.openacademia.org`

The appearance of the publication list can be customized by choosing from a variety of stylesheets.

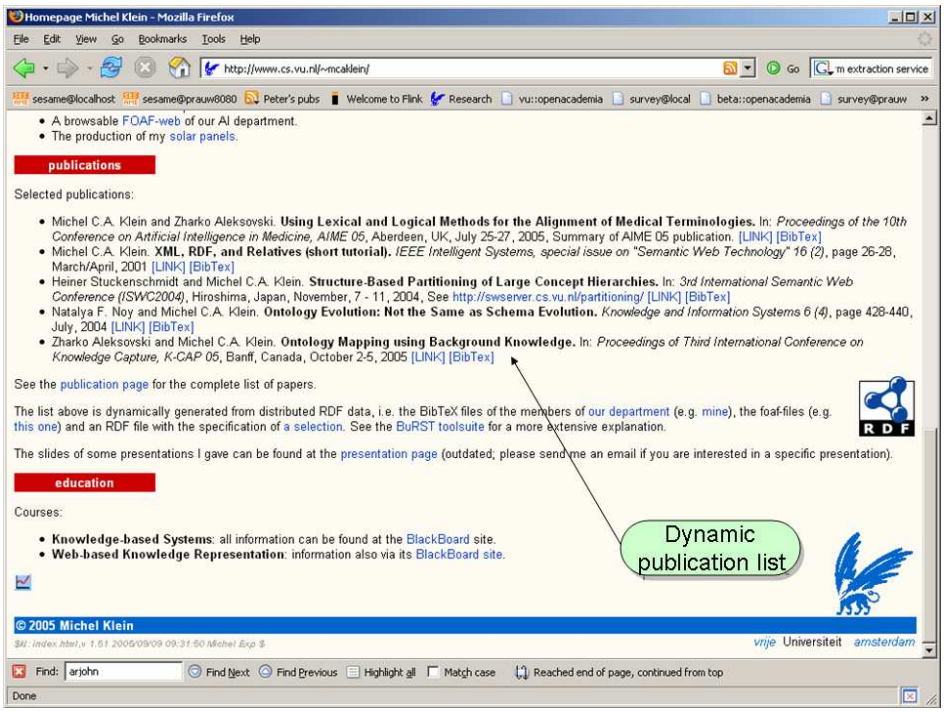


Figure 7.8: Dynamically generated publication list inserted to a web page using remote syndication.

More interestingly, one can also generate an RSS feed from the collection. Adding such an RSS feed to a homepage allows visitors to subscribe to the publication list using any RSS news reader. Whenever a new publication is added, the subscribers of the feed will be notified of this change through their reader (*information push*).

A number of generic tools are available for reading and aggregating RSS information, including browser extensions, online aggregators, news clients and desktop readers for a variety of platforms. Mozilla Firefox also natively supports RSS feeds as the basis for creating dynamic bookmark folders (see Figure 7.9). These folders refresh their contents from an RSS feed whenever the user opens them.

The RSS feeds of openacademia are RDF-based and can also be consumed by any RDF aware software such as Piggy Bank browser extension.[22] Piggy Bank allows users to collect RDF statements linked to Web pages while browsing through the Web and to save them for later use.

[22] `simile.mit.edu/piggy-bank/`

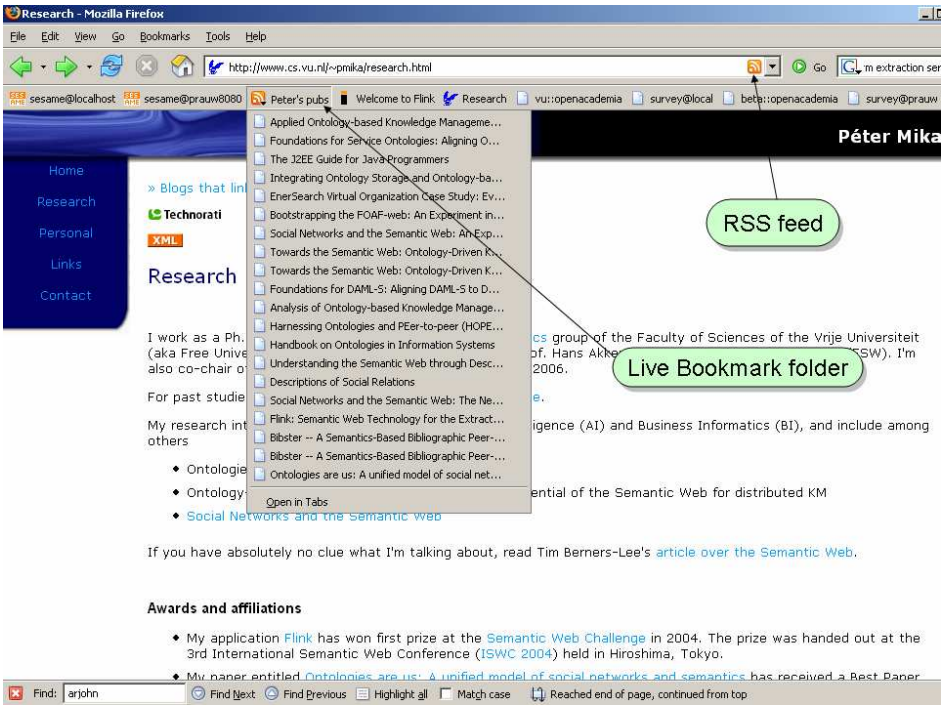


Figure 7.9: Modern browsers such as Mozilla Firefox have built in RSS feeders that allow to subscribe to RSS feeds attached to webpages. Using the Live Bookmark feature of Firefox it is also possible to save a publication list as a bookmark folder that automatically refreshes itself.

Research groups can install their own openacademia server. Members of the research group can submit their publications by creating a FOAF profile pointing to the location of their publication collection. What the system provides is the possibility to create a unified group publication list and post it to a website similarly to personal lists. (Needless to say, groups can have RSS feeds as well.)

There is also an AJAX based interface for browsing and searching the publication collection (see Figure 7.10). This interface offers a number of visualizations. For example, the important keywords in the titles of publication matching the current query can be viewed as a *tagcloud*[23], where the size of the tags shows the importance of the keyword. It is also possible to browse the co-authorship networks of researchers using the same interactive applet used by Flink. Another interactive visualization shows publication along a timeline that can be scrolled using the mouse (see Figure 7.11).[24]

Keywords or tags can be added to publications using the features of BibTeX or EndNote. The system also extracts keywords automatically from titles of publications.

[23] `http://en.wikipedia.org/wiki/Tag_cloud`

[24] This time-based visualization uses the Timeline widget developed by the SIMILE project. See `http://simile.mit.edu/timeline/`

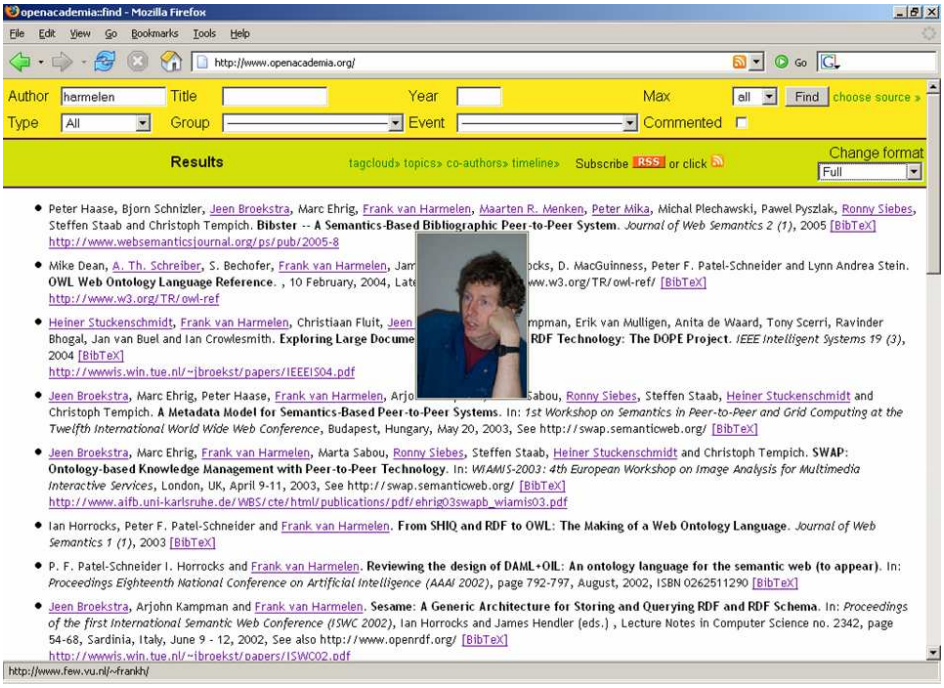


Figure 7.10: The AJAX-based query interface of openacademia builds queries and displays the results. If the underlying data source includes FOAF data even the researchers' photos are shown (with links to their homepage).

Lastly, openacademia connects to blog search engines in order to import blog comments about publications.

7.3.2 System design

The architecture of openacademia follows the same design as Flink: in the middle of the architecture is an RDF repository that is filled with a variety of information sources and queried by a number of services (see Figure 7.12).

The difference lies in the dynamics of the two systems. Flink is filled with data every two or three months in a semi-automated fashion. openacademia repositories refresh their content every day automatically.[25] In case a publication feed is generated from a single BibTeX or EndNote file the entire process of filling and querying the repository is carried out on the fly. In this case we use an in-memory repository that is discarded after the answer has been served.

[25]The frequency of updates can be configured. The advantage of a daily update is that a change in any of the sources propagates to the triple store within 24 hours. The disadvantage is that updates take computing resources and time during which the repository is unavailable.

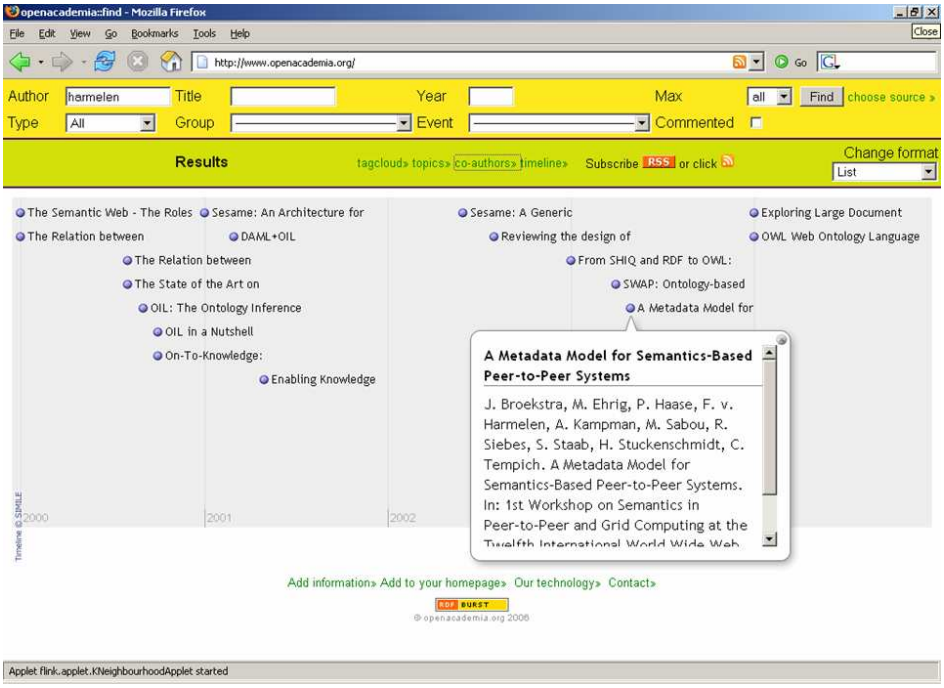


Figure 7.11: Interactive time based visualization using the Timeline widget.

Information collection

For obtaining metadata about publications, we rely on the BibTeX and EndNote formats commonly in use in academia worldwide. We ask authors to include a BibTeX file with their own publications on a publicly accessible part of their website. For most authors in the sciences this does not require additional work, as they typically maintain their own web space and have such a file at hand. Further, researchers fill out a form to create a basic FOAF profile, which contains at least their name and the location of their references. Needless to say, this step can be skipped in case the researcher already has a FOAF profile.

We use the Elmo crawler to collect such profiles. As mentioned before, the crawler can be restricted to a domain, which is useful for limiting the data collection to the domain of an institute. The BibTeX files are translated to RDF using the BibTeX-2-RDF service,[26] which creates instance data for the "Semantic Web Research Community" (SWRC) ontology. A simple extension of the SWRC ontology was necessary to preserve the sequence of authors of publications. To this end we defined the *swrc-ext*:*authorList* and *swrc-ext*:*editorList* properties, which have *rdf*:*Seq* as range, comprising an ordered list of authors.

[26] See `http://www.cs.vu.nl/~mcaklein/bib2rdf/`.

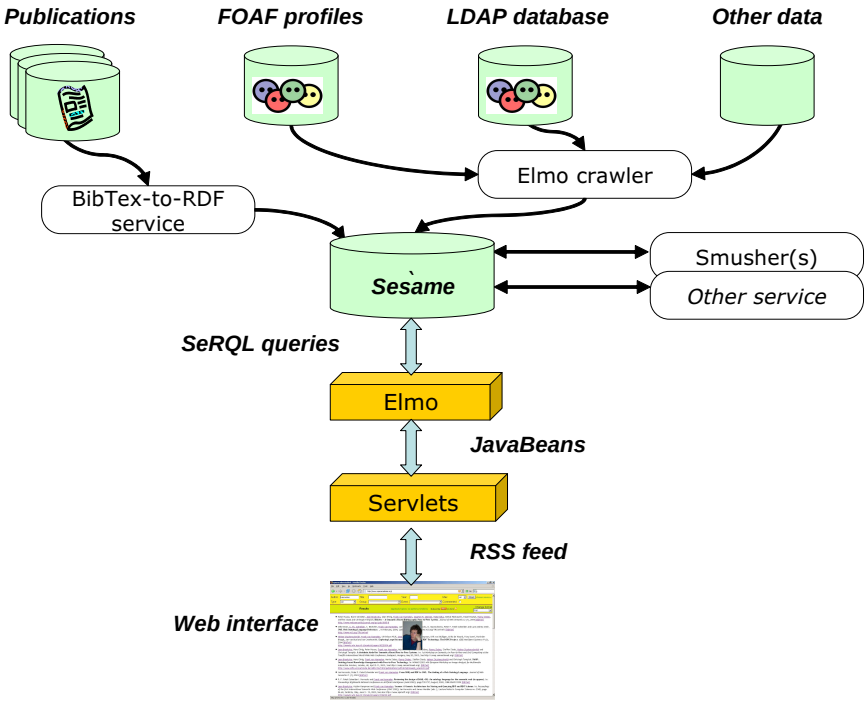


Figure 7.12: The architecture of openacademia.

At the Vrije Universiteit, Amsterdam we have also implemented a service that dynamically queries the local LDAP database and represents the contents as FOAF profiles. Most importantly, the LDAP database contains information about group membership, but it also contains real names and email addresses. We do not reveal the email addresses of employees, but use a hash of the email address as identifier. By relying on the LDAP database for this information we delegate the task of maintaining user data to the existing infrastructure in the department i.e. the user account administration.

Storage and aggregation

As our system is designed to reuse information in a distributed, web-based environment, we have to deal with the arising semantic heterogeneity of our information sources. Heterogeneity effects both the schema and instance levels.

Similarly to Flink, the schemas used are stable, lightweight web ontologies (SWRC and FOAF) and their mapping causes no problems. Heterogeneity on the instance level arises from using different identifiers in the sources for denoting the same real world objects. This certainly effects FOAF data collected from the Web (where typically each personal profile also contains partial descriptions of the friends of the individual), but

also publication information, as the same author or publication may be referenced in a number of BibTeX sources.

We use the Elmo smusher framework to match *foaf*:*Person* instances based on name and inverse-functional properties. Publications are matched on a combination of properties. In the current system, we look for an exact match of the date of the publication and a tight fuzzy match of the title. Matching publications based on authors is among the future work.

The instance matches that we find are recorded in the RDF store using the *owl*:*sameAs* property. Since Sesame does not natively support OWL semantics at the moment, in earlier versions of the system we expanded the semantics of this single property using Sesame's custom rule language. These rules expressed the reflexive, symmetric and transitive nature of the property as well as the intended meaning, namely the equality of property values. The disadvantage was that these rules added a large number of statements by assigning all the equivalent resources the same the set of properties. On the upside these rules are executed by the custom inferencer during uploads, which means that queries are fast to execute. In the current solution we apply the approach outlined in Section 6.4.4, i.e. a combination of forward chaining and backward chaining.

As in the case of Flink, semantic technology allows us to infer additional knowledge from the data. For example, we can add a rule to our knowledge base which states that the co-authors of publications are persons who know each other. The generated information can be used to enrich the personal profiles of individuals, but also to further disambiguate individuals based on their friends.

Presentation

After all information has been merged, the triple store can be queried to produce publications lists according to a variety of criteria, including personal, group or publication facets. The online interfaces helps users to build such queries against the publication repository. The queries are processed by another web-based component, the publication web service.

In order to appreciate the power of a semantics-based approach, it is illustrative to look at an example query. The query in Figure 7.13, formulated in the SeRQL language, returns all publications authored by the members of the Artificial Intelligence department in 2005. This department is uniquely identified by its homepage. (The *foaf*:*homepage* property is inverse functional according to the FOAF vocabulary, i.e. a certain URL uniquely identifies a given group.)

Note first that the successful resolution of this query relies on the schema and instance matching described in the previous section. The query answering requires publication data as well as information on department membership and personal data from either the LDAP database or the self-maintained profile of the researcher. This points to a clear advantage of the Semantic Web approach: a separation of concerns. Researchers can change their personal profiles and update their publication lists without the need to consult or notify anyone. Similarly, the maintainer of the departmental LDAP database can add and remove individuals as before. All this information will be connected and merged automatically.

```
SELECT DISTINCT pub
FROM {group} foaf:homepage
{<http://www.cs.vu.nl/ai/>}; foaf:member {person},
{pub} swrc:year {year}; swrc:author {person}
WHERE year="2004"
USING NAMESPACE
  foaf = <http://xmlns.com/foaf/0.1/>,
  swrc = <http://swrc.ontoware.org/ontology#>
```

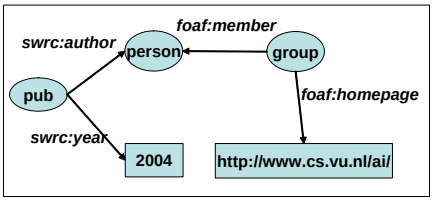


Figure 7.13: Sample SeRQL query and its graphical representation.

The publications that match this query necessarily come from different sources, i.e. from the collections of personal publications of individual researchers. Publications that have been co-authored by members of the department will appear in a number of BibTeX sources, possibly with different spellings of the author names and providing a different set of metadata (e.g. one source may describe only the authors and the title, while another may mention the title and year, but not the authors). Identifying matching publications is the task of the publication smusher introduced before, which is thus also essential to correctly answering this query.[27]

The publish service takes a query like the one shown above, the location of the repository, the properties of the resulting RSS channel and optional style instructions as parameters. In a single step, it queries the repository, performs the post-processing and generates an RSS channel with the publications matching the query.

The resulting channel appears as an RSS 1.0 channel for compatible tools while preserving RDF metadata. The method of mixing publication metadata into an RSS channel is described in the BuRST specification[28]. The BuRST format —illustrated in Figure 7.14— is based on an explicit rule of RSS processing, namely that RSS parsers (many of them working on the XML tree of data) should ignore all information that they do not understand. This means that if the general structure of the RSS document is preserved (i.e. all mandatory elements are present and written in the correct order) existing parsers will turn a blind eye to the additional metadata attached to RSS items. In our case, we attach publication metadata to RSS items using the *burst*:*publication* property. This is the only property defined by the BuRST specification.

The presentation service can also add XSL stylesheet information to the RSS feed, which allows to generate different HTML layouts (tables, short citation lists or longer descriptions with metadata). The HTML output can be viewed with any XSLT capable browser and it can be tailored even further by adding a custom CSS stylesheet (for changing colors, font styles etc.)

Stylesheets are also used to generate the XML input of the Timeline widget. One can even reproduce a BibTeX or EndNote representation of the publication feed by applying the appropriate stylesheets.

[27] The query still requires some post processing as we might encounter duplicate publications that have different URIs; SeRQL is an RDF query language with no support for OWL equivalence. The *DISTINCT* keyword guarantees only that the URIs returned are unique.

[28] `http://www.cs.vu.nl/~pmika/research/burst/BuRST.html`

```
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:swrc="http://swrc.ontoware.org/ontology/ontoware#"
   xmlns:foaf="http://xmlns.com/foaf/0.1/"
   xmlns:dc="http://purl.org/dc/elements/1.1/"
   xmlns:burst="http://xmlns.com/burst/0.1/">

 <rss:channel rdf:about="http://www.cs.vu.nl/~pmika/burst.rdf">
  <rss:title>Peter Mika's publications</rss:title>
   <rss:link>http://www.cs.vu.nl/~pmika/research/pub.rdf</rss:link>
   <rss:description>
    Semantic Web related publications authored by Peter Mika.
   </rss:description>
  <rss:items>
    <rdf:Seq>
       <rdf:li rdf:resource="http://www.cs.vu.nl/~pmika/burst#1" />
       <rdf:li rdf:resource="http://www.cs.vu.nl/~pmika/burst#2" />
    </rdf:Seq>
  </rss:items>
  <rdfs:seeAlso rdf:resource="http://www.cs.vu.nl/~mcaklein/pub.rdf" />
 </rss:channel>

 <rss:item rdf:about="http://www.cs.vu.nl/~pmika/burst#1">
   <rss:title>Bootstrapping the FOAF-Web: An Experiment
          in Social Network Mining
   </rss:title>
   <rss:link>http://www.w3.org/2001/sw/Europe/events/foaf-galway/papers/fp/bootstrapping_the_foaf_web/</rss:link>
   <rss:description>
      Bootstrapping is a problem that affects all applications of the
      Semantic Web, including the network of interlinked Friend-of-a-Friend (FOAF)
      profiles known as the FOAF-web. In this paper we introduce a hybrid system ...
   </rss:description>
   <dc:subject>Semantic Web</dc:subject>
   <burst:publication>
      <swrc:InProceedings>
         <swrc:title>Bootstrapping the FOAF-Web: An Experiment in Social Network Mining</swrc:title>
         <swrc:author>
           <foaf:Person rdf:ID="PeterMika">
              <foaf:name>Peter Mika</foaf:name>
              <foaf:mbox_sha1sum>ffe33bbe8be2a2123f0adb793e61a6d84ae9a739</foaf:mbox_sha1sum>
              <rdfs:seeAlso rdf:resource="http://www.cs.vu.nl/~pmika/foaf.rdf" />
           </foaf:Person>
         </swrc:author>
         <swrc:year>2004</swrc:year>
      </swrc:InProceedings>
   </burst:publication>
 </rss:item>
 ...
```

Figure 7.14: Example of a BuRST channel.

7.4 Discussion

It is just a few years ago that the development of a Semantic Web application still required the kind of effort represented by European projects such as On-To-Knowledge, which combined the best minds in academia with the development prowess of industry [Davies et al., 2003]. Only a few select developers (the *übergeeks* of the community) possessed the kind of technical skills to do the same independently.

However, the effort required to develop applications for the Semantic Web has significantly dropped in the past years. The W3C opened the way to stable software development by finishing the standardization of the RDF and OWL ontology languages (2004) and the SPARQL query language and protocol (2006), which together provide for the interoperability of semantic applications and services. Open source tools such as the

Sesame storage facility have also significantly matured both in terms of robustness and scalability.

Programming tools such as the Elmo API contribute to this development by reducing the learning and adaptation required from the average Java developer. Elmo shields the developer from the low level details of representation by offering static object models for working with common web ontologies such as FOAF, RSS and Dublin Core. It integrates with Sesame to provide additional services such as caching.

Elmo also provides the common components required by most Semantic Web applications. As we have seen on the example of Flink and openacademia one of these common tasks is the crawling (scuttering) of RDF data from the data sources. In both cases, these data included RDF documents from the Web as well as structured data from the sources that were not originally intended for these applications (email archives, publication databases). The dynamic collection of data from a variety of sources is a typical characteristic of Semantic Web applications as this is the scenario where semantic technology brings the greatest benefits. Crawling data from interconnected data sources is the task of the Elmo scutter used both by Flink and openacademia.

Another common task in working with heterogeneous data sources is the aggregation of instance data, which can be automated by using the extensible smusher framework of Elmo. As we have discussed in the Chapter on data aggregation, instance reasoning often requires procedural components as the kind of knowledge required cannot be fully represented in ontology languages (and thus this task cannot be solved by reasoners alone).

Flink and openacademia are examples of Semantic Web applications that address very different problems: data collection for network analysis and publication metadata management. Nevertheless, they re-use many of their components by building on Elmo, which allows the developer to focus on the specific task of these applications (the so called *business logic*) and reduce the complexity of the code to a minimum.

As a demonstration of how to apply this technology for e-science, and in particular network analysis, we will use the Flink system as our tool for data collection and management in our study on the Semantic Web research community (see Chapter 9. But first, in order to justify our use of electronic data, we will take a closer look at the relationship between real-world networks and online networks in a scientific setting.

Part III

Case Studies

Chapter 8

Evaluating electronic data extraction for network analysis

The Internet and computer communication networks in general have been extensively studied as indicators for social networks from as early as 1996. Several internet studies suggest that internet interaction replaces the traditional role of social networks [Otte and Rousseau, 2002, Wellman et al., 1996]. Taking this observation to the extreme, network researchers with a background in physics or computer science study online networks as the equivalents of real world networks. On the other hand, network analysts with a social science background apply an extreme caution and most often still treat electronic communication networks and online communities as a separate field of investigation where different theories and methods apply than in the study of real world networks.

Nevertheless, surprisingly little is known about the exact relationship between real world networks and their online reflections. In particular, the question is: to what extent electronic data obtained from the Web reveal social structures such as those mentioned by actors when questioned by specific network questionnaires?

In the following we limit our attention to data generated by methods of social network mining from the Web as introduced in Chapter 4 and discussed in [Mika, 2005a, Matsuo et al., 2006]. It is likely that different forms of electronic data could serve as a source for obtaining information about different types of social relations. For electronic data such as email traffic that represent message exchanges between individuals a higher degree of correspondence with social network data is plausible while for others such as web logs a more distant perspective seems to be warranted. Social network mining from the Web based on co-occurrence is an interesting method as it is likely to produce evidence of ties of different strengths based on the variety of the underlying data.

Further, email and online collaboration is likely to be used in different ways in different social settings. Bearing in mind the limitations of this method, we have chosen

members of a research organization[1] as the subjects of our study. Scientific communities are particularly well-suited to be studied based on their online presence due to the amount and variety of information that is present online, including personal homepages, information on research collaborations in the form of projects and publications, descriptions of events (conferences and workshops) etc.

We choose to evaluate electronic data extraction against the results of a survey method, which is the dominant approach to data collection for network analysis. Standard questionnaires are preferred in theory testing for their fixed structure, which guarantees the comparability of results among test subjects and across studies. Various forms of asking for one's social relations have been tested through the years for reliability. The fixed structure of questionnaires also allows to directly extract relational data and attributes for processing with network analysis tools and statistical packages. Questionnaires are also minimally invasive and can be easily mailed, emailed or administered online.

In the following we compare the results of our social network mining methods with the outcomes of a study we have conducted using a common type of network questionnaire. Note that we will call the results of our survey the golden standard, reflecting the viewpoint of network analysis. However, we can already project one of the most important limitations of our analysis: with any discrepancy we may find between the results of online and off-line analysis the difference could be attributed to limitations of either of the two. While network questionnaires are dominant, they are hardly infallible and by treating them as a benchmark we may try to match their imperfection. (And vice versa: if we were to treat online data as a benchmark for evaluating survey methods we would try to reproduce their errors.)

Our study is unique as we have no knowledge of any evaluation of network extraction methods using real world data. Thus our main contribution is the description of how such an evaluation can be carried out and the questions it raises.[2] Second, we will use this evaluation to compare different network mining methods, fine tune their parameters and measure the effect of disambiguation techniques. Lastly, we will give a sense of how good the results of these extraction methods are by comparing network measures computed on our web-based networks with the same measures calculated on the graphs from our survey.

We expect this evaluation to be relevant for anyone who is considering to use data extracted from the Web as a proxy for real world data. Web-based extraction requires significantly less effort to execute than data collection using surveys or observation. Further, the results can be scaled to any number of participants and the data collection can be repeated at any number of times, even retrospectively, provided that the community to be investigated had a significant web presence at the time. As more and more communities move more of their activities online, we expect that web mining will become a preferred source of data for many network experiments in the future.

[1] We will use the more specific term research organization to denote scientific groups (communities) active in a formal organization.

[2] Unfortunately, we cannot make our data set public due to privacy restrictions. This limitation holds for most real world experiments in network analysis. In fact, we have no knowledge of any publicly available data sets that we could have used for our evaluation. Nevertheless, it would be highly desirable to have such a data set as a benchmark for evaluating network analysis methods in a completely transparent way.

8.1 Differences between survey methods and electronic data extraction

In the above we have already mentioned some of the key advantages of electronic data extraction, which mostly have to do with the differences in how data are collected. In the following we focus on the outcomes and summarize the potential sources of disagreement between the results we obtain from a typical network survey compared to the results of our method of mining the Web for social network data.

This list will serve as a guideline when looking for possible explanations for the discrepancies in the results. We will explicitly test for some of these differences as potential sources of disagreements and leave some for future work.

Note that some of these points are specific to our method of network extraction or the specific research setting.[3] Nevertheless, this list can also serve as a checklist for anyone considering similar comparisons, possibly in different settings using different methods of network data extraction.

- Differences in what is measured
  - What is not on the Web cannot be extracted from the Web, which limits the scope of extraction. Also, these data can be biased in case part of the community to be studied is better represented on the Web than other parts. This is similar to the sampling problem of survey methods: one needs to take care that the data collected allows to build a balanced representation of the underlying social structure.
  - Our network extraction method is likely to find evidence for different kinds of relationships resulting in what is called a multiplex network. These relationships are not easily entangled although some progress can be made by applying machine learning to disambiguate relationships. Matsuo et al. demonstrate this method by learning typical features of Web pages that characterize certain relationships in the research domain [Matsuo et al., 2006]. They report good results in distinguishing between co-authors, colleagues and co-participation relationships.

    We address this problem differently: we measure a number of relationships in our survey and use these data to understand the composition of relationships we find on the Web (see Section 8.6).
  - The equivalent problem in survey methods is the difficulty of precisely formulating those questions that address the relationship the researcher actually wants to study. This is a hard, if not the hardest problem as shown by the attention that is paid to this question in most network studies. For example, the key network question in the General Social Survey (GSS) of the United States was originally formulated to ask respondents about the persons with whom they discuss *personal matters* [McPherson et al., 2006]. This was

[3] We have discussed the potential sources of errors in network extraction in Chapter 4, but we repeat these points here for the sake of completeness.

later changed to the same question asking about *important matters* because in a separate study it was found that respondents varied greatly in their understanding of personal matters and sometimes interpreted this term in very narrow ways [Ruan, 1998]. Even such a minor change, however, complicates the comparison between results obtained using the original question and the new formulation [McPherson et al., 2006].

- Errors introduced by the extraction method
  - There are errors that affect the extraction of particular cases. Homonymy affects common names (e.g. *J. Smith* or *Xi Li*), but can be reduced somewhat by adding disambiguation terms to queries. Synonymy presents a problem whenever a person uses different variations of his or her name. Different variations of first names (e.g. *James Hendler* vs *Jim Hendler*), different listing of first and middle names, foreign accentuation, different alphabets (e.g. Latin vs. Chinese) etc. can all lead to different name forms denoting the same person.

    In the following, we will address this problem by experimenting with various measures that could predict if a particular name is likely to be problematic in terms of extraction. We can test such measures by detecting whether the personal network of such persons is in fact more difficult to extract than the networks of other persons.
  - Another class of errors is likely to affect all names with equal probability. An example of such a systemic error is the accidental co-occurrence of two names on a web page. Further, even if intended not all co-occurrences carry equal weight e.g. many co-occurrences are likely to be duplicates.

    Such systemic errors or noise are likely to be reduced by means of large numbers and are altogether less threatening as they affect all cases with equal probability.
- Errors introduced by survey data collection
  - Unlike network mining from the Web, surveys almost never cover a network completely. Although a response rate lower than 100% is not necessarily an error, it does require some proof that either the non-respondents are not significantly different from the respondents with respect to the survey or that the collected results are so robust that the response from the non-respondents could not have affected it significantly.
  - The respondents are not likely to be equally co-operative either. There are most likely differences in the level of cooperativeness and fatigue. Some of these factors can be measured and checked in order to make sure that the responses by the less cooperative or more fatigued population are not significantly different from the rest [McPherson et al., 2006]. In small scale studies the situation is even more problematic as the non-respondents and the respondents are likely to be clustered as the subjects discuss the matter of the survey and influence each other in responding to it or not.

– The mere fact of observation can introduce a bias. At best this bias affects all subjects in an equal manner.
– Not only the type of relationship that is considered by the respondent but also the recall of contacts is affected by the way a question is formulated (see Section 8.3) and how it is placed in the sequence of questions [McPherson et al., 2006].

8.2 Context of the empirical study

We have collected network data on the social networks of the 123 researchers working at the Department of Computer Science of the Vrije Universiteit, Amsterdam in September 2006. We have received contact information about the researchers and an approval for our study from the head of the Department.

The Department is organized is six Sections of various sizes, which are in decreasing order of size: Computer Systems (38), Artificial Intelligence (33), Information Management and Software Engineering (22), Business Informatics (17), Theoretical Computer Science (9) and Bioinformatics (4).[4] The Sections are further divided internally into groups[5], each led by a professor. Researchers in each section include part- or full-time PhD students, postdocs, associate and full professors, but the study excluded master students, support staff (scientific programmers) and administrative support.

Note that this research organization is by nature different from a research community such as the Semantic Web community introduced earlier in that is not primarily bound by a broad research interest but rather a shared affiliation with an institution. Nevertheless, this affiliation is rather loose as most within the organization consider the level of the department as a purely administrative level. Thus the overall image is close to that of a research community: in both cases relationships between individuals are largely driven by specific research interests or affiliation with a smaller group[6] rather than their common relationship with the organization or research community. (As it turns out, the network of the organization is even less centralized than the research community of the Semantic Web.)

We have chosen this community as a subject of our study because the author is a member of the Business Informatics group of the Department. This position allowed us to directly address the participants of the study and most likely resulted in a higher response rate than it would have been possible otherwise. On the downside, some participants felt that this study should not have been carried out by "one of their own" as they did not feel comfortable with providing personal information to a colleague, even with the assurance that only statistics would be computed from the data and that the data would not be used for management purposes.[7] In effect, nine people have voted out of the study. From the remaining 114 researchers we have collected 79 responses (a response rate of 64%), with

[4] For more information on the research of the department, please visit `http://www.cs.vu.nl`

[5] These are the *leerstoelgroepen* in Dutch.

[6] Members of the same group are typically also co-located in the same parts of the building.

[7] No doubt most of the participants did not realize how much information about their networks is available online.

above average response rates in the BI group (88%) and the closely interlinked AI (79%) group.

8.3 Data collection

We collected personal and social information using a custom-built online survey system. An online survey offers several advantages compared to a paper questionnaire:

- Easy accessibility for the participants. The participants did not need to be physically present. They were sent a user name and a password in email and could log on anytime.
- Greater flexibility in design, allowing for a better survey experience. Using an electronic survey it is possible to adapt questions presented to the user based on the answers to previous questions. This reduces the time to complete the survey and thus diminishes non-response.
- Easier processing for the survey administrator. Our system recorded electronic data directly in RDF using the FOAF-based semantic representations discussed in Chapter 6. As we already had the components for reading FOAF data and exporting it in the formats supporting by SNA packages, the data required no post-processing. Also, the system did part of the error checking as the participants were filling out the survey (e.g. checking that all mandatory fields were filled out).

There are a number of electronic survey tools available on the Web either for free or against a small payment. Unfortunately, these solutions allow very little customization for either error checking or for implementing survey logic. Further, as these off-the-shelf tools are not specifically designed for network surveys, they would have produced data in formats that are difficult to post-process (typically Excel sheets).

The survey is divided over several pages. The first page asks the participant to enter basic personal information: his or her full-time or part-time status, age, years at the organization, name of the direct supervisor and research interests. The second and third pages contain standard questions for determining the level of self-monitoring and the extent someone identifies with the different levels of the organization. These control variables were not used in the current evaluation.

The fourth page asks the participant the select the persons he or she knows from a complete list of Department members. This question is included to pre-select those persons the participant might have any relationship with. The next page asks the participant to specify the nature of the relationship with the persons selected. In particular, the participant is suggested to consider six types of relationships and asked to specify for each person which type of relationship applies to that person.

The six types of relations we surveyed were advice seeking, advice giving, future cooperation, similarity perceptions, friendship, and adversarial ties. The first three questions assessed instrumental ties, whereas the last three questions represented affective ties and general perceptions of similarity.

The advice network was operationalized following [Krackhardt, 1990] and [Ibarra and Andrews, 1993]. The respondents were asked to look through an alphabetical list of all employees and check the names of the people "who you regularly approach if you have a work-related problem or when you want advice on a decision you have to make". Advice giving was operationalized analogously with a formulation "who regularly approaches you…". Future cooperation inquired about preferences for future work by asking "who would you like to work with in the future".

As for the similarity perceptions and affective ties, we followed the operationalization of [Mehra et al., 1998] for the identity network asking the respondents to check the names of the people "who you consider especially similar to yourself". In assessing friendship, we asked to mark those people "whom you consider a personal friend, e.g., a person you like to spend breaks with or engage in social activities". This question was found to reliably distinguish a "friendly relationship" from a "friend" in a validation study, conducted at a dialysis and nursing department of a Dutch hospital [van de Bunt, 1999]. Moreover, the question assessing the friendship network in the study of Mehra, Kilduff, & Brass was formulated in a similar vein [Mehra et al., 2001]. Following Baldwin, Bedell, & Johnson [Baldwin, Timothy T. et al., 1997] and van de Bunt [van de Bunt, 1999] we assessed the negative relations asking respondents to check names of the people "with whom you have a difficult relation, e.g., you cannot get along with this person".

We used the roster method instead of a free recall to collect social network data (cf. [Wasserman et al., 1994]). The alternative survey method of name generators carries the risk that participants forget to mention some of their contacts. (They name those that first come to mind.) In order to limit the measurement error, respondents were not restricted to a fixed number of nominations [Holland and Leinhardt, 1973].

We believe that the two step process of pruning the list of participants before presenting relationship choices was beneficial because it put much less strain on the participant than the traditional questionnaire method where the participant fills out a fixed-size matrix. Such a matrix would be very large and sparse in our case, which increases the likelihood of errors, the time to fill out the survey and consequently the risk of abandonment. In fact, most of our survey users informally reported that the survey in the current form was very smooth and easy to "click through", costing about 15 minutes of their time.

Upon completion of the last page, the survey software stored the results in a Sesame RDF store [Broekstra et al., 2002], saved the data on backup and notified the survey administrator in email.

8.4 Preparing the data

The networks we have collected each contained 123 nodes corresponding to the equal number of survey participants. In a first step, we have removed all non-respondents from these networks, reducing the number of nodes to the 79 respondents. The number of nodes with edges in each of the networks is lower, since not all respondents provided information about all of their networks. Note also that this step also reduced the number

Graph	Advice seeking	Advice giving	Friendship	Troubled relation	Similarity	Future work
Nodes after non-respondent removal	79	79	79	79	79	79
Nodes w/edges	74	58	70	18	60	74
Edges	363	290	447	40	323	546
Edges after non-respondent removal	226	194	298	21	221	361
Edges after direction removal	197	161	229	20	193	282

Table 8.1: Statistics of our networks from the survey.

of edges in the networks since respondents may have mentioned non-respondents and these edges were removed. Non-respondents were also removed from the graph obtained from web mining: this graph contained edges also for those who did not respond to our survey.

Next, we removed directionality from our survey networks and our web-based network. It is unlikely that the directionality measured by our social network extraction method would correspond directly to any of the measured dimensions. Further, the network measures we generally compute do not take directionality into account. Table 8.1 shows the basic statistics about our collected data before and after the preprocessing.

8.5 Optimizing goodness of fit

In order to prepare the comparison with our web-based networks, we had to filter the nodes and edges of this network and remove the directionality in this case as well.

Filtering of the web-based network requires to specify cut-off values for two parameters: the minimal number of pages one must have on the Web to be included (pagecount) and the minimal strength of the relationships (strength). Recall that the first parameter is used to exclude individuals with too few pages to apply web mining, while the second parameter is used to filter out ties with too little support (thus one is a criterium on the nodes to be included and the other is a criterium on the edges to be included, see Chapter 4).

Figures 8.1 and 8.2 show the distribution of these values. Note that while the strength values show a smooth power-law distribution, there are two different scaling regimes for the page counts. Typically PhD students have less than 1000 pages on the Web, while post-docs and more senior department members have over 10000 pages. Note also that unlike in the case of the Semantic Web community, we do not use any disambiguation term, which results in some outliers due to famous namesakes (e.g. Michel Klein).

We mention that filtering is either carried out before removing directionality or one needs to aggregate the weights of the edges going in different directions before the edges

can be filtered. Some alternatives are taking the sum, average, maximum or minimum of the values.

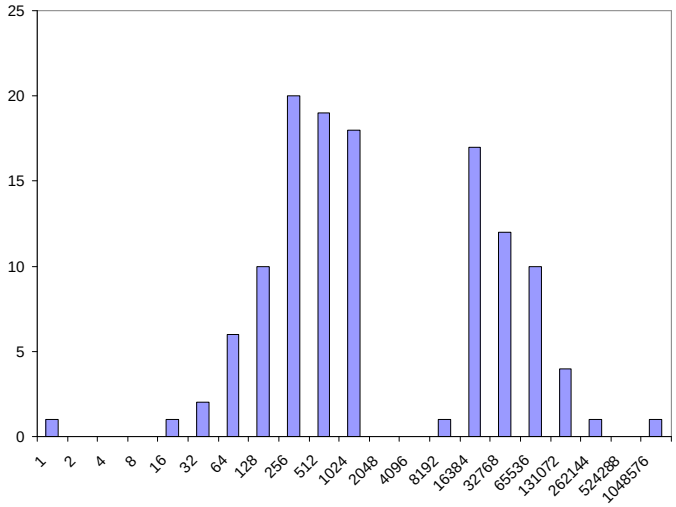


Figure 8.1: Histogram for the number of web pages per individual. Note that the x-axis is on a logarithmic scale.

Finding the appropriate parameters for filtering can be considered as an optimization task where we would like to maximize the similarity between our survey networks and the extracted network. For this optimization we need to choose a similarity measure.

We can consider relationship extraction as an information retrieval task and apply well-known measures from the field of information retrieval. Let's denote our graphs to be compared as $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$. Some of the measures to be considered are listed in Table 8.2. Precision, recall and the F-measure are common measures in information retrieval (see e.g. [van Rijsbergen, 1979]), while the Jaccard-coefficient is also used for example in UCINET [Borgatti et al., 2002].

Once we have chosen a measure, we can visualize the effect of the parameters on the similarity using surface plots such as the one shown in Figures 8.3. These figures show the changes in the similarity between the advice seeking network and the network obtained from web mining as we change the pagecount and strength thresholds (the two horizontal axes show the parameter settings and the vertical axis shows the resulting similarity). As expected, for symmetric measures such as the F-measure this is a convex surface: extremes along both axis result in zero similarity. Figures 8.4 and 8.5 show

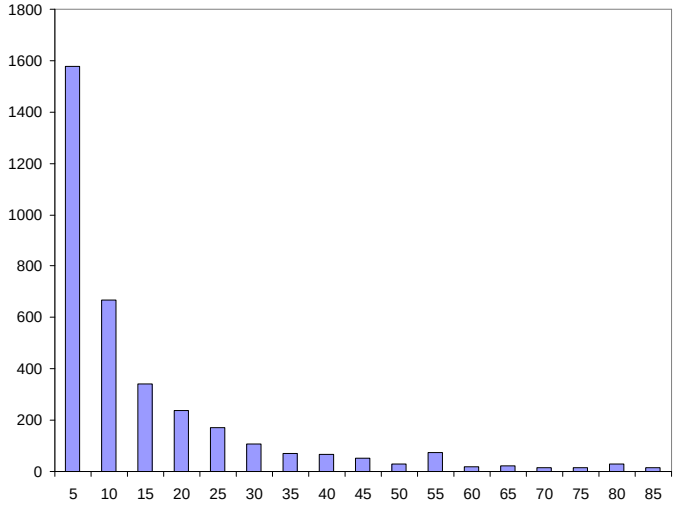


Figure 8.2: Histogram for the strength of relationships based on the web extraction method.

the asymmetric precision and recall measures, which help us understand why this is the case. Setting the thresholds to zero results in high recall (82%) but low precision (21%). Conversely, setting the thresholds high results in high precision (approximating 100%) but low recall (reaching 0%) corresponding to the extreme case where all nodes and edges have been filtered out up to the last edge.

The F-measure, which is the harmonic mean of precision and recall, has a single highest peak (optimum) and a second peak representing a different trade-off between precision and recall. As it is the case that the mining graph at this second peak contains more nodes and edges (higher recall, lower precision), it might be preferable over the optimum depending on the goal of the application. To accommodate differences in the importance of precision and recall, it is also possible to apply a weighted version of the F-measure.

In general, we note that it seems easier to achieve high recall than high precision. This suggests the possibility of a two-stage acquisition process where we first collect a social network using web mining and then apply a survey in which we ask respondents to remove the incorrect relations and add the few missing ones. Such a pre-selection approach can be particularly useful in large networks where listing all names in a survey would result in an overly large table. Further, subjects are more easily motivated to correct lists than to provide lists themselves.

Jaccard-coefficient	$\frac{\lvert E_1 \cap E_2 \rvert}{\lvert E_1 \rvert + \lvert E_2 \rvert - \lvert E_1 \cap E_2 \rvert}$
Precision	$\frac{\lvert E_1 \cap E_2 \rvert}{\lvert E_1 \rvert}$
Recall	$\frac{\lvert E_1 \cap E_2 \rvert}{\lvert E_2 \rvert}$
F-measure	$2 * \frac{(\lvert E_1 \cap E_2 \rvert)^2}{\lvert E_1 \rvert * \lvert E_1 \cap E_2 \rvert + \lvert E_2 \rvert * \lvert E_1 \cap E_2 \rvert}$

Table 8.2: Similarity measures for graphs based on edge sets

8.6 Comparison across methods and networks

Our benchmark survey data also allows a direct comparison of methods for social network mining. In this case we compare the best possible results obtainable by two (or more) methods, i.e. we choose the parameters for each method separately such that some similarity measure is optimized.

We have subjected to this test our benchmark method of co-occurrence analysis and the method based on average precision (see Chapter 4). Figure 8.6 shows a comparison of these two methods in terms of the lowest precision, highest recall and highest F-measure they can achieve on any of our six networks.

The results confirm our intuition that the average precision method produces higher precision, but lower recall resulting in only slightly higher F-measure values. By looking at the numbers across networks, we can see that the easiest to predict are the advice seeking, advice giving and future work networks. This is true for both methods, i.e. it seems to be a feature of the data, rather than the method of extraction. The possibility to predict future work relations might be a bit surprising considering that the question behind it is about the future. However, respondents were not specifically instructed to exclude current work relations.

In fact, it is likely that the relationships we extract from the Web reflect a number of underlying relationships, including those we asked in our survey and possibly others we did not. To measure to what extent each of our surveyed relationships is present on the Web it would be possible to perform a p* analysis, where we assume that the Web-based network is a linear combination of our survey networks (see Equation 8.1).

$$\lambda_1 S_1 + \lambda_2 S_2 + \lambda_3 S_3 + \lambda_4 S_4 + \lambda_4 S_4 + \lambda_5 S_5 = W \tag{8.1}$$

The analysis would then return the coefficients of this linear equation, i.e. the weights with which the individual survey networks contribute to the Web-based network. Notice that for this we need to fix the parameters of the Web-based network. In fact, we can consider this as an optimization task where we try to maximize the fit between the survey networks and the Web-based network by adjusting the weights of the survey networks instead of changing the parameters of the Web-based network as we have done previously. This kind of analysis is among the future work.

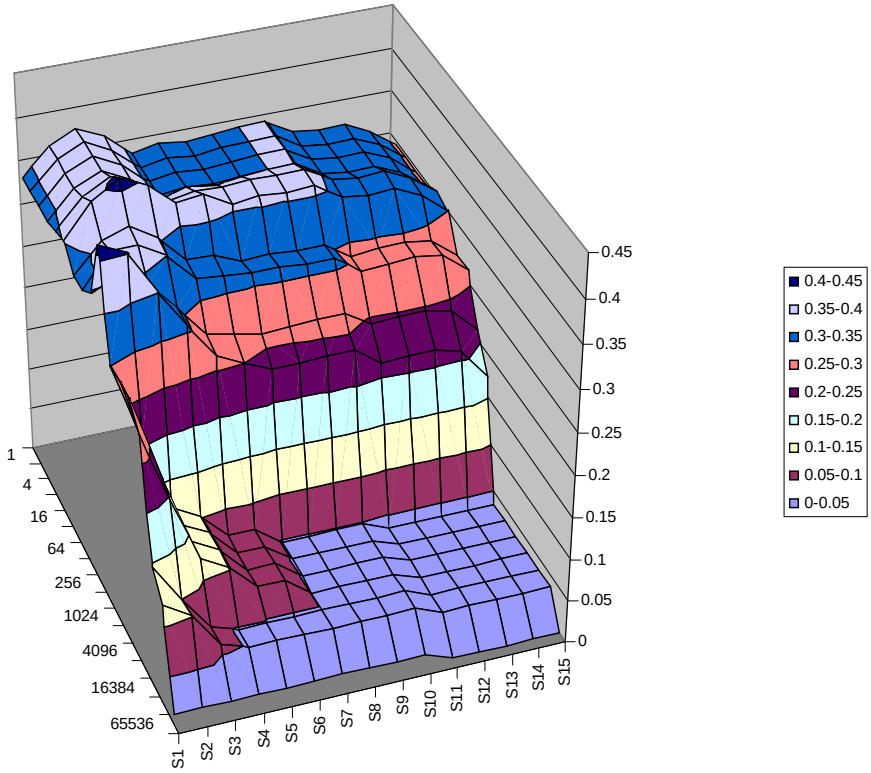


Figure 8.3: Similarity between the advice seeking network and the network obtained from the Web using the F-measure. The similarity (plotted on the vertical, z axis) depends on the value of the two parameters of the algorithm.

8.7 Predicting the goodness of fit

In Chapter 4 we have suggested the influence of a number of factors on the success of extracting social networks from the Web. Many of these factors are personal. For example, the amount of information available about a person or the commonality of someone's name on the Web. In our current study, we also find that the number of pages on the Web mentioning someone varies widely in the community, partly affected in fact by the problem of namesakes (either many namesakes or a single famous namesake).

Based on the attribute data we have collected from our survey (and other attributes we can compute) it is interesting to investigate whether some of these factors can indeed be linked to the success of obtaining the person's social network from the Web. Trying to predict a-priori the goodness of fit between an individual's real social network and its Web-based image has important practical significance: if we find measures that help us to predict when the extraction is likely to fail we can exclude those individuals from the

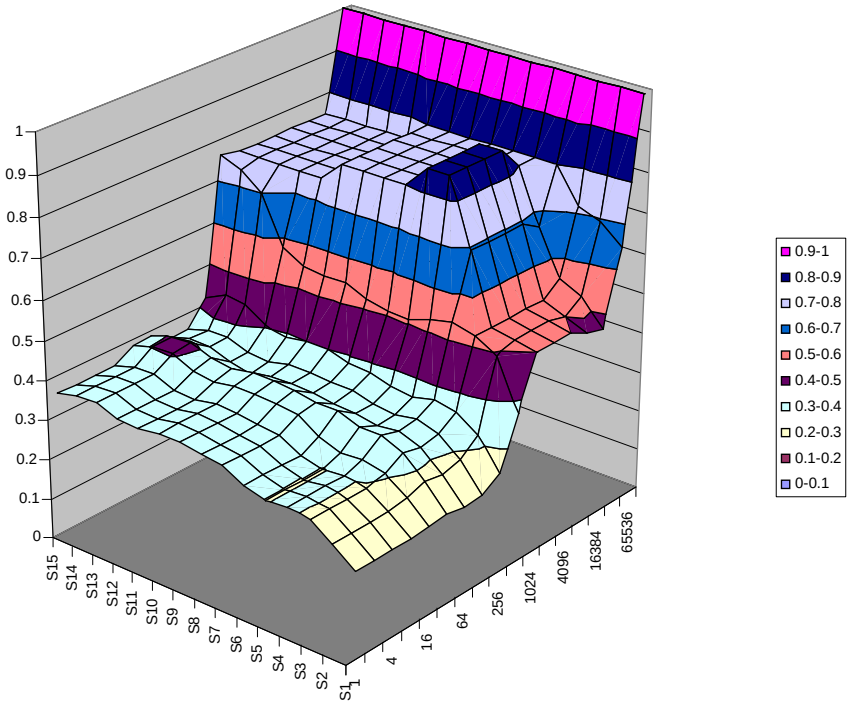


Figure 8.4: Similarity between the advice seeking network and the network obtained from the Web using the precision measure. The precision (plotted on the vertical, z axis) depends on the value of the two parameters of the algorithm.

Web-based extraction and try other methods or data sources for obtaining information about their relations.

To carry out this kind of analysis, we have to measure the similarity between personal networks from the survey and the Web and correlate it with attributes of the subjects. (Thus, in contrast to the previous section, we compute similarity measures on ego-networks.) In this kind of study, we work with a single network from our survey and a mining network with fixed parameters. (As described previously, we fix the parameters of extraction such that the overall fit between the survey and the network from the Web is maximized.)

The attributes we consider are those from our survey, e.g. the number of relations mentioned (*surveydegree*), the age of the individual and the number of years spent at the VU (variables *age* and *entry*). We also look at Web-based indicators such as the number of relations extracted (*miningdegree*) and the number of pages for someone's name, which we recode based on its distribution by taking the logarithm of the values (*pagecount*). Last, we experimented with measures for name ambiguity based on the existing literature on name disambiguation [Bekkerman and McCallum, 2005,

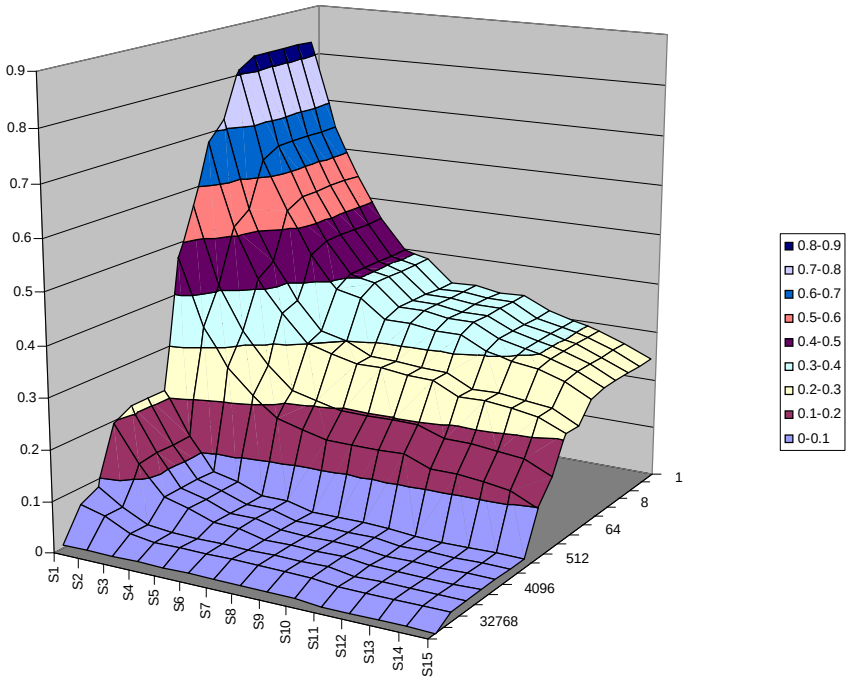


Figure 8.5: Similarity between the advice seeking network and the network obtained from the Web using the recall measure. The recall (plotted on the vertical, z axis) depends on the value of the two parameters of the algorithm.

Bollegala et al., 2006]. Both of these works apply clustering methods on the results returned by the search engine in order to predict how many persons or personalities[8] are found in the result set and which pages belong to which individual. These measures try to estimate the web presence of the particular individual we are looking for:

- NC1: Jaccard-coefficient between the first name and the last name. The idea behind this measure is to predict how much a first name and a last name belong together. The expectation is that the larger this quotient, the least common the name is.

- NC2: The ratio of the number of pages for a query that includes the full name and the term *Vrije Universiteit* divided by the number of pages for the full name only. This is a measure of the number of pages that are certainly about the correct

[8] A person may have multiple contexts of activity, even if we consider the professional activities alone. For example, Noam Chomsky is known both as a linguist and a prolific political writer. In an application where we want to distinguish his networks in these different contexts, we would prefer a disambiguation module that is able to separate the pages related to these two activities even if they belong to the same person.

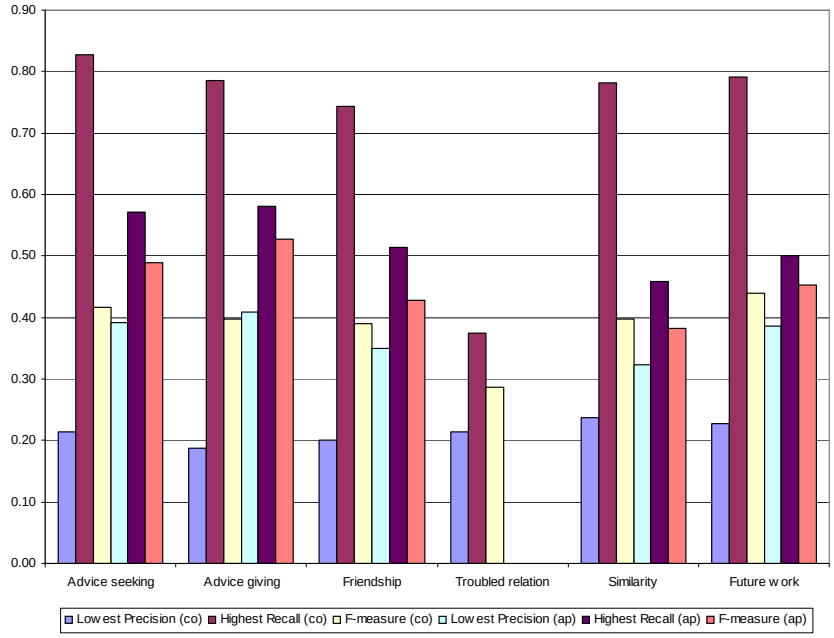


Figure 8.6: Similarity between the advice seeking network and the network obtained using the recall measure. The recall (plotted on the z axis) depends on the value of the two parameters of the algorithm.

individual versus the total number of pages. It is expected that the higher this value, the easier it is to retrieve the correct individual.

In general, we find that the more respondents are mentioned by someone, the higher the precision of the extraction. There can be two alternative explanations to this. First, it is possible that some respondents are mentioning more of their social network than others and the more relations they name the higher the precision goes. Second, it is possible that respondents with larger social networks have a correspondingly larger web presence, which makes it easier to extract their correct relations. We also trivially find that the more relations are extracted from the Web for any individual, the lower the precision and the higher the recall.

Interestingly, none of the survey attributes has a direct influence on the result, even though age and entry are correlated with the number of web pages as expected. The NC1 measure has no significant effect. On closer inspection, a problem with this measure is that it assigns high scores also to persons who have a common first name and last name, but whose combination of first and last name is rare. The measure is also unable to distinguish between name commonality (e.g. Li Ding) and famous namesakes (e.g. George Bush). Somewhat surprisingly, the NC2 measure has a negative effect on the

		surveydeg	miningdeg	age	entry	na_1	na_vu	pc_recode	fmeasure	precision	recall
surveydeg	Pearson	1	0.667	-0.035	-0.004	-0.058	-0.038	**0.361**	**0.390**	**0.355**	0.148
	Sig. (2-tailed)		0.000	0.764	0.975	0.617	0.737	**0.003**	**0.023**	**0.034**	0.357
	N	79	79	77	77	77	79	**65**	**34**	**36**	41
miningdeg	Pearson		1	-0.090	-0.022	-0.072	0.128	0.152	-0.206	**-0.370**	**0.798**
	Sig. (2-tailed)			0.434	0.848	0.534	0.260	0.227	0.241	**0.026**	**0.000**
	N		79	77	77	77	79	65	34	**36**	**41**
age	Pearson			1	**0.671**	0.169	-0.001	**0.325**	0.015	-0.133	-0.136
	Sig. (2-tailed)				**0.000**	0.146	0.994	**0.009**	0.933	0.448	0.404
	N			77	**77**	75	77	**63**	34	35	40
entry	Pearson				1	0.072	0.029	**0.332**	-0.003	-0.189	-0.057
	Sig. (2-tailed)					0.540	0.799	**0.008**	0.985	0.276	0.729
	N				77	75	77	**63**	34	35	40
na_1	Pearson					1	-0.140	**0.250**	0.155	0.089	0.025
	Sig. (2-tailed)						0.223	**0.048**	0.390	0.611	0.880
	N					77	77	**63**	33	35	40
na_vu	Pearson						1	-0.114	**-0.498**	-0.277	0.226
	Sig. (2-tailed)							0.367	**0.003**	0.103	0.155
	N						79	65	**34**	36	41
pc_recode	Pearson							1	0.019	-0.030	-0.159
	Sig. (2-tailed)								0.920	0.872	0.354
	N							65	30	32	36
fmeasure	Pearson								1	**0.694**	0.128
	Sig. (2-tailed)									**0.000**	0.471
	N								34	**34**	34
precision	Pearson									1	-0.111
	Sig. (2-tailed)										0.520
	N									36	36
recall	Pearson										1
	Sig. (2-tailed)										
	N										41

**. Correlation is significant at the 0.01 level (2-tailed).

*. Correlation is significant at the 0.05 level (2-tailed).

Figure 8.7: Correlations between personal attributes and the similarity between the personal networks from the survey and the Web.

F-measure. The explanation could be that respondents who have a high percentage of their mentioning linked to the Vrije Universiteit produce a higher recall which results in lower precision (at an already high level of recall).

8.8 Evaluation through analysis

Typically, networks from surveys or the Web are used as raw data for computing complex measures of Network Analysis. What we are really interested in is thus the extent to which a Web-based network can replace a survey when used in analysis. It is likely that if the network extraction is optimized to match the results of the survey it will give similar results in analysis. However, we will see that a 100% percent match is not required for obtaining relevant results in applications: most network measures are statistical aggregates and thus relatively robust to missing or incorrect information.

Group-level analysis, for example, is typically insensitive to errors in the extraction of specific cases. Figures 8.9 and 8.8 show the structure of our advice seeking network and the optimized mining network, including the group affiliation of individuals. (Names have been replaced with numbers for privacy reasons.) The macro-level social structure

of our department can be retrieved by collapsing this network to show the relationships between groups using the affiliations or by clustering the network. (These functions are available for example in Pajek.) By applying these procedures to the two networks they reveal the same underlying organization: two of the groups (the AI and BI sections) built close relationships with each other and with the similarly densely linked Computer Systems group. The explanation for the first observation is that several members of the two groups are involved in similar research on the topic of the Semantic Web. (This mismatch between the informal and formal organization is also very well recognized by the researchers themselves.) The remaining groups are either very small (Theoretical Informatics, Bioinformatics) or very loosely connected both internally and to the other groups (IMSE).

Our experiments also show the robustness of centrality measures such as degree, closeness and betweenness. For example, if we compute the list of the top 20 nodes by degree, closeness and betweenness we find an overlap of 55%, 65% and 50%, respectively. While this is not very high, it signifies a higher agreement than would have been expected: the correlation between the values is 0.67, 0.49, 0.22, respectively. The higher correlation of degrees can be explained by the fact that we optimize on degree when we calibrate our networks on precision/recall.[9]

In general, we argue that the level of similarity between our survey and Web-based networks is high enough that it is unlike that we find a network effect based on the Web data that would not hold if we were to use the survey network. The rationale is similar to arguing for sampling in social sciences. When working with a sample instead of the entire population we might obfuscate some significant effects. However, if the sampling is done in a uniform manner it is unlikely that we introduce a significant effect just through the process of sampling.

8.9 Discussion

The Web is rapidly becoming one of the most important sources of data for network analysis. Not only the adoption of the Web is taking off, but also the Web itself has been turning into a place of active socialization through technologies such as social networking, blogging, instant messaging, and platforms collaborative work on software and content. The availability of electronic data presents an unprecedented opportunity for network science in conducting large scale studies using dynamic (time-bound) data. Further, these data can be either collected as is or it can be easily extracted through well-known methods of web mining. The resulting data set can be directly manipulated by machines, which further lowers the costs of network studies.

Before Web data can be subject to analysis, however, network analysis needs to investigate the extent to which Web-based data can replace the input from traditional survey and observation methods of data collection. In this Chapter we have addressed this question by providing methodological guidance for evaluation and argued that in the particular domain of scientific communities electronic data that can be collected on the Web

[9] One might play with the idea of optimizing fit based on these network measures, e.g. selecting the parameters of network mining such that the correlation between centrality measures is maximized.

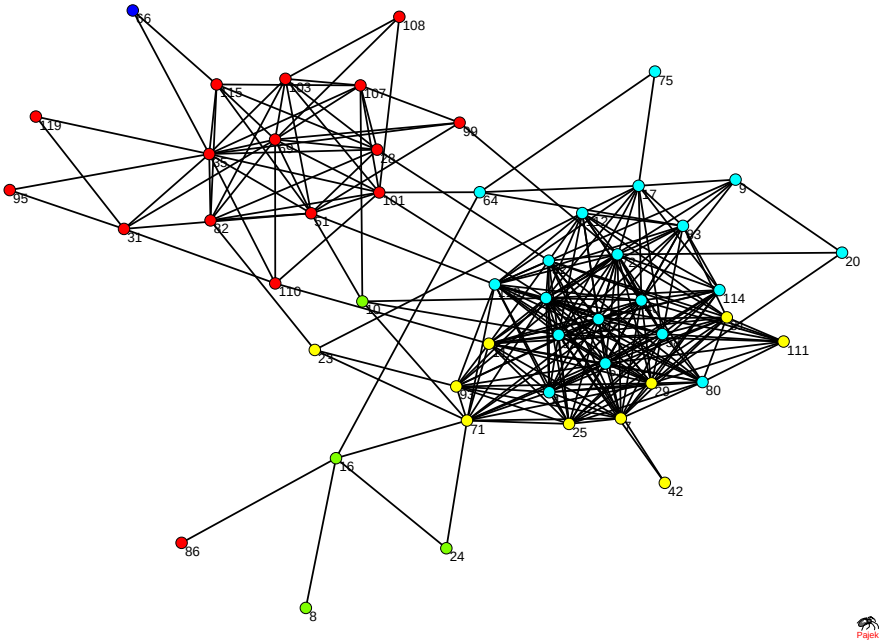


Figure 8.8: The network from Web mining (after optimizing for maximum similarity with the advice seeking network). Colors indicate different Sections within the Department.

is a close enough reflection of real world networks to use it in place of survey data as the input of network analysis.

We have noted that the disagreement between our alternate methods could be explained by either differences in what is being measured or imperfections in either the survey or the web extraction method of data collection. Although it is the dominant method of data collection, one should in fact be cautious in considering survey data as a golden standard as it can lead to reproduce the imperfections of the survey method. We have tested for some of the explanations that could explain existing differences and proposed some as future work.

Evaluations similar to ours could be (and should be) carried out with a variety of network data in order to prove the robustness of our methods across domains. For example, network mining methods could be applied to the extraction of corporate networks, which could be more easily tested against existing databases on joint ventures and other forms co-operation as used in [Lemmens, 2003].

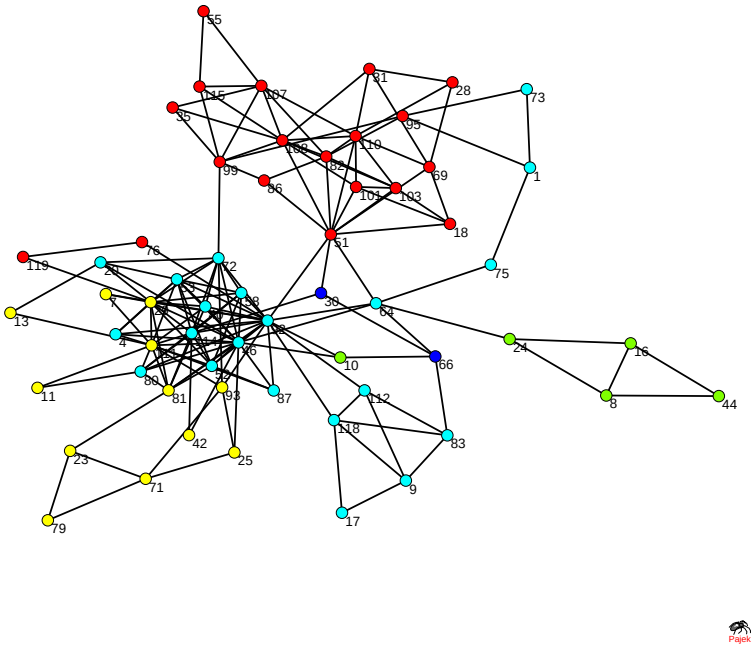


Figure 8.9: The advice seeking network of researchers. Colors indicate different Sections within the Department.

Chapter 9

Semantic-based Social Network Analysis in the sciences

In social studies of science, research fields have been considered as self-organized communities, also referred to as invisible colleges [Crane, 1971]. Recently smaller groups with shared research interests have been addressed as virtual teams [Ahuja and Carley, 1999, Ahuja et al., 2003]).

Both notions relate to the observation that social connectivity is relevant to many key aspects of research. Conceptualizing research fields as either self-organized communities or virtual teams also allows us to study them using network analysis methods of social science. In past works, various features of the social networks of researchers have proved useful in explaining scientific performance on the individual and organizational level.[1]

The informal nature of scientific communities, however, has also made it difficult to obtain network data for analysis. Traditionally, information about the informal social structure of scientific communities is gathered through labor-intensive methods of data collection, e.g. through interviews or network questionnaires. Alternatively, researchers from deSolla Price [deSolla Price, 1965] to Barabási [Barabási et al., 2002]) have relied on more tangible evidence of formal work relations such as co-authoring and co-citation of scientific publications or investigated co-participation in research projects [Heimeriks et al., 2003, Grobelnik and Mladenic, 2002]. For obtaining much of these data and for measuring scientific performance, most authors have relied on commercially available databases of journal articles, patent- and project grants.

More recently, the use of electronic data extraction is gaining ground in the study of networks. While traditional survey or interview methods are limited in the size of networks and the number of measurements (time-points), electronic data enable large scale, longitudinal studies of networks. In many cases, electronic data also break the reliance on commercial providers of information, lowering the costs of access. This

[1] Studies at the intersection of network analysis and scientometrics appear either in the journal Scientometrics (Akadémiai Kiadó, co-published with Springer Science+Business Media B.V.) or in the Social Network Analysis literature (see Chapter 3.

process is visible, for example, in the unfolding clash between freely consultable online publication databases based on Information Extraction technology (such as Citeseer[2] and Google Scholar[3]) and those maintained by the publishers themselves.

Scientific communities in high technology domains such as Artificial Intelligence or Bioinformatics are among the ones that lend themselves most naturally to be studied through their online presence, due the openness of the (academic) research environment and their use of advanced communication technology for knowledge sharing in emails, forums and on the Web. For example, email communication in research and standardization settings are the source of social networks in [Gloor et al., 2003] and [Adamic and Adar, 2005], while other studies extract social networks from the content of web pages [Kautz et al., 1997, Mori et al., 2004] or —somewhat less successfully— by analyzing the linking structure of the Web [Heimeriks et al., 2003]. As the first to publish such a study, Paolillo and Wright offer a rough characterization of the FOAF[4] web in [Paolillo and Wright, 2004].

The availability of a multitude of information sources provides the opportunity to obtain a more complete view of the social network of a scientific community, thereby it increases the robustness and reliability of research designs. In particular, by decreasing our reliance on the single sources of information our findings become less prone to the errors in the individual sources of information. However, the availability of a number of data sources also poses a previously unmet challenge, namely the aggregation of information originating from heterogeneous information sources not primarily created for the purposes of network analysis.

We meet this challenge by offering methodological contributions that also allow us to carry out novel theoretical investigations into network effects on scientific performance.

First, we propose the use of semantic technology in the management of social network data, in particular the semantics-based aggregation of information from heterogeneous information sources. Our system extracts information from web pages, emails and online collections of publications. Semantic technology allows us to uniquely identify references across these sources and merge network data. We will show the validity of this approach by relating network positions to real world status in a scientific community and using our data in proving the well known hypothesis of Ronald Burt with respect to the positive effect of structural holes on innovativeness [Burt, 1995].

Second, we advance the theory of network effects on scientific performance by going beyond a structural analysis of our networks and incorporating the effects of cognitive diversity in ego networks. Our analysis of the potential content of relationships is enabled by our method of extracting the research interests of scientists from electronic sources. In particular, we will look at the content of relations and the effects of cognitive (dis)similarity. We hypothesize that cognitive diversity in the ego network of researchers will be positively related to their performance, especially for newcomers, juniors researchers entering the field.

[2] `http://citeseer.ist.psu.edu/`

[3] `http://scholar.google.com`

[4] FOAF is a popular semantic-based format for describing user profiles and social networks on homepages. For more information, please see the FOAF project at `http://www.foaf-project.org/`

In the following, we begin by introducing the context of our study, the Semantic Web research community. In Section 9.2 we introduce our first contribution, our system for extracting and aggregating social network information from various electronic information sources. In Section 9.3, we formalize our hypotheses concerning the effects of social and cognitive networks on scientific performance and test these hypotheses using the data collected. We summarize our work in Section 9.4.

9.1 Context

The context of our study is the community of researchers working towards the development of the Semantic Web, an extension of the current Web infrastructure with advanced knowledge technologies that have been originally developed in the Artificial Intelligence (AI) community. The idea of the Semantic Web is to enable computers to process and reason with the knowledge available on the World Wide Web. The method of extending the current human-focused Web with machine processable descriptions of web content has been first formulated in 1996 by Tim Berners-Lee, the original inventor of the Web [Berners-Lee et al., 1999].

The Semantic Web has been actively promoted since by the World Wide Web Consortium (also led by Berners-Lee), the organization that is chiefly responsible for setting technical standards on the Web. As a result of this initial impetus and the expected benefits of a more intelligent Web, the Semantic Web has quickly attracted significant interest from funding agencies on both sides of the Atlantic, reshaping much of the AI research agenda in a relatively short period of time.[5]

As the Semantic Web is a relatively new, dynamic field of investigation, it is difficult to precisely delineate the boundaries of this network.[6] For our purposes we have defined the community by including those researchers who have submitted publications or held an organizing role at any of the past International Semantic Web Conferences (ISWC02, ISWC03, ISWC04) or the Semantic Web Working Symposium of 2001 (SWWS01), the most significant conference series devoted entirely to the Semantic Web. We note that another commonly encountered way of defining the boundary of a scientific community is to look at the authorship of representative journals (see e.g. [Heimeriks et al., 2003]). However, the Semantic Web has not had a dedicated journal until 2004 and still most Semantic Web related publications appear in AI journals not entirely devoted to the Semantic Web.

The complete list of individuals in this community consists of 608 researchers mostly from academia (79%) and to a lesser degree from industry (21%). Geographically, the community covers much of the United States, Europe, with some activity in Japan and Australia (see Figure 2.4). As Figure 2.5 shows, the participation rate at the individual

[5]Examples of some of the more significant projects in the area include the US DAML program funded by DARPA and a number of large projects funded under the IST initiative of the EU Fifth Framework Programme (1998-2002) and the Strategic Objective 2.4.7 of the EU Sixth Framework Programme (2002-2006).

[6]In fact, it is difficult to determine at what point does a new research concept become a separate field of investigation. With regard to Semantic Web, it is clear that many of the scientists involved have developed ties before their work on the Semantic Web, just as some of the research published in the Semantic Web area has been worked out before in different settings.

ISWC events have quickly reached the level typical of large, established conferences and remained at that level even for the last year of data (2004), when the conference was organized in Hiroshima, Japan. The number of publications written by the members of the community that contain the keyword "Semantic Web" has been sharply rising since the beginning.

9.2 Methodology

Our methodology combines existing methods of email and web mining with novel, semantic-based techniques for storing, aggregating and reasoning with social network data. Flink, our self-implemented semantic software supports the complete process of data collection, storage and visualization of social networks based on heterogeneous sources of electronic data (see Section 7.2).

The idea of semantic-based representations of user profiles and social networks originates from technology-aware online communities (the blog world), who were among the earliest adopters of Semantic Web technology. In particular, the format of the Friend-of-a-Friend (FOAF) project was quickly adopted by users, because it allowed to store profiles and social network information linked to the homepages of users, breaking the reliance on profit-oriented social networking services such as Friendster or Orkut. The use of semantic technology also offered the advantage of extensibility: adding new properties to the profiles could be done without breaking compatibility. An example of this is the Speaks-Reads-Writes ontology[7] for describing what languages the user can read, speak and write.

While semantic technology has been quickly adopted by online communities, it has been left largely unnoticed in the social sciences, despite important benefits for the management of social network data. As Flink demonstrates, semantic technology allows us to map the schema of our information sources and to find correspondences among the instances. The use of standard semantic languages for the representation of social science data makes it possible to use generic Semantic Web tools and infrastructure for editing, storing, querying and reasoning with our data. Lastly, the semantic data store is the basis for a web-based user interface for browsing the data set, computing social network statistics and exporting the networks and the results of the computations.

Figure 9.1 shows a high level overview of the architecture of the Flink system. The three layers of the system, concerned with data acquisition, representation and visualization, will be introduced separately in the following sections.

We end this Section with a discussion about the use of electronic data for social network analysis. We address both the benefits of electronic data as well as the concerns that can be raised in terms of the reliability of our methods.

9.2.1 Data acquisition

The first layer of the Flink system is concerned with data acquisition. Flink makes use of four different types of knowledge sources: text-based HTML pages from the web, FOAF

[7] `http://www.schemaweb.info/schema/SchemaInfo.aspx?id=48`

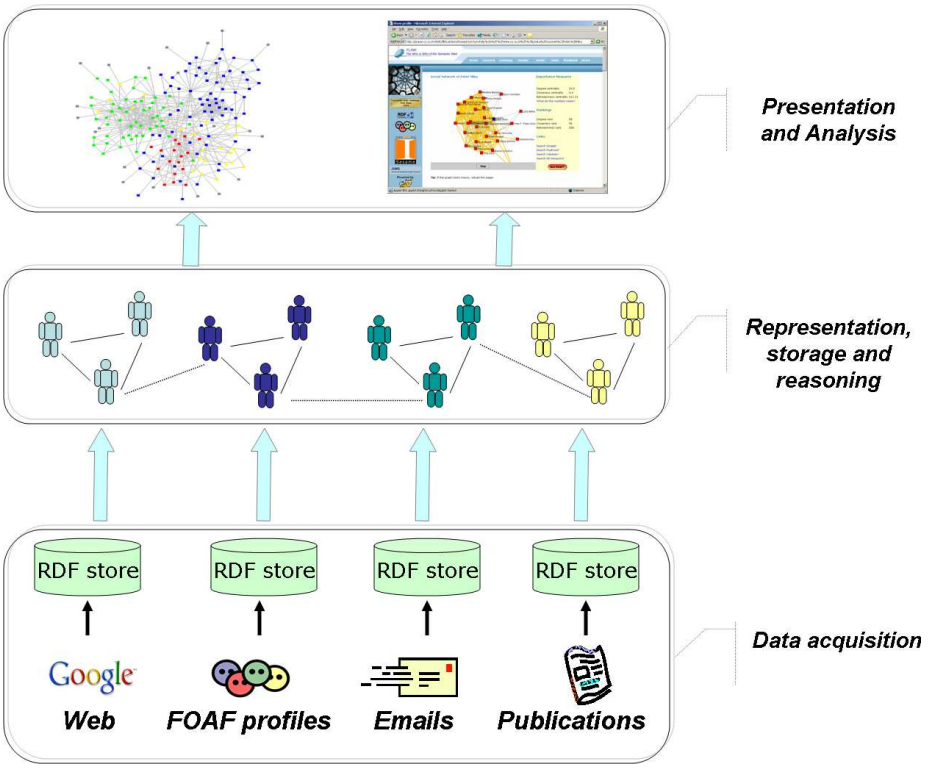


Figure 9.1: An overview of the architecture of the Flink system.

profiles, public collections of emails and bibliographic data. Information from the different sources is extracted in different ways as described below. In the final step, however, all the data gathered by the system is represented in a semantic format (RDF), which allows us to store heterogeneous data in a single knowledge base and apply reasoning (see the following Section).

The web mining component of Flink extracts social networks from web pages using a co-occurrence analysis technique introduced in Chapter 4. Although the technique has also been applied before in the AI literature to extract social networks for the automation of referrals (see [Kautz et al., 1997]), to our knowledge this is the first time that the output of the method is subjected to network analysis.

Given a set of names as input, the system uses the search engine Google to obtain the number of co-occurrences for all pairs of names from the membership list of the ISWC community. (The term "(Semantic Web OR ontology)" is added to the queries for disambiguation.) We filter out individuals whose names occurs less then a certain threshold, because in their case the extracted relationships would have very low support.

The absolute strength of association between individuals is then calculated by dividing with the page count of a single individual. In other words, we calculate the fraction of pages where both names are mentioned compared to all pages where an individual is mentioned.[8] The resulting associations are directed and weighted. We consider such an association as evidence of a tie if it reaches a certain predefined threshold. In our experiments this minimum is set at one standard deviation higher than the mean of the values, following a "rule of thumb" in network analysis practice.

The web mining component of Flink also performs the additional task of associating individuals with domain concepts. In our study of the Semantic Web community, the task is to associate scientists with research interests. (The list of terms characterizing research interests has been collected manually from the proceedings of ISWC conferences.) To this end, the system calculates the strength of association between the name of a given person and a certain concept. This strength is determined by taking the number of the pages where the name of an interest and the name of a person co-occur divided by the total number of pages about the person. We assign the expertise to an individual if this value is at least one standard deviation higher than the mean of the values obtained for the same concept.[9] This is different from the more intricate method of Mutschke and Quan Haase, who first cluster keywords into themes, assign documents to themes and subsequently determine which themes are relevant for a person based on his or her publications [Mutschke and Haase, 2001].

[8] We have also experimented with normalization using the Jaccard-formula, but we found that it gives unsatisfactory results if there is a large discrepancy between the web-representation of two individuals. This is the case, for example, when testing the potential relationship between a Ph.D. student and a professor.

[9] Note that we do not factor in the number of pages related to the concept, since we are only interested in the expertise of the individual relative to himself. By normalizing with the page count of the interest the measure would assign a relatively high score —and an overly large number of interests— to individuals with many pages on the Web. We only have to be careful in that we cannot compare the association strength across interests. However, this is not necessary for our purposes.

We can map the cognitive structure of the research field by folding the bipartite graph of researchers and research interests.[10] In the resulting simple graph (shown in Figure 9.5) vertices represent concepts, while an edge is drawn between two concepts if there are at least a minimal number of researchers who are interested in that particular combination of concepts. Note that this is different from the commonly used simple co-word analysis. By using a two-step process of associating researchers to concepts and then relating concepts through researchers we get a more accurate picture of the scientific community. Namely, the names of researchers disambiguate the meaning of words in case a word is understood differently by different authors.

Information from emails is processed in two steps. The first step requires that the emails are downloaded from a mail server and the relevant header information is extracted. In a second step, the individuals found in the collection are matched against the profiles of the members of the target list to filter out relevant profiles from the collection. (See Section 9.2.2.)

Although not used in the current experiment, FOAF profiles found on the Web can also be used as an information source. First, an RDF crawler (scutter) is started to collect profiles from the Web. A scutter works similar to an HTML crawler in that it traverses a distributed network by following the links from one document to the next. Our scutter is focused in that it only collects potentially relevant statements, i.e. those containing FOAF information. The scutter also has a mechanism to avoid large FOAF producers that are unlikely to provide relevant data, in particular blog sites[11]. Once FOAF files are collected, the second step again involves filtering out relevant profiles.

Lastly, bibliographic information is collected in a single step by querying Google Scholar with the names of individuals (plus the disambiguation term). From the results we learn the title and locations of publications as well as the year of publication and the number of citations where available.[12] An alternative source of bibliographic information (used in previous versions of the system) is the Bibster peer-to-peer network [Haase et al., 2004], which allows to export bibliographic information in an RDF-based format.

9.2.2 Representation, storage and reasoning

All information collected through the data acquisition layer is represented in RDF (Resource Description Framework) using the FOAF vocabulary (see Chapter 5).

In terms of data management for the social sciences, RDF is a key technology for aggregating information from heterogeneous information sources. The first step in this process is to express all information using a common representation, i.e. RDF. Personal

[10] Bipartite graphs of people and concepts are known as affiliation networks (two-mode networks) in SNA practice. Two-mode networks can be used to generate two simple networks, showing associations between concepts and people [Wasserman et al., 1994].

[11] The overwhelming presence of these large sites also make FOAF characterization difficult. See [Paolillo and Wright, 2004]. We ignore as we do not expect many Semantic Web researchers to maintain blogs and the amount of information would make it difficult to work with the data.

[12] Note that it is not possible to find co-authors using Google Scholar, since it suppresses the full list of authors in cases where the list would be too long. Fortunately, this is not necessary when the list of authors is known in advance.

information and social networks are described in FOAF, emails are expressed in a proprietary ontology, while publication metadata is expressed in terms of the SWRC ontology.

After normalizing syntax, the next step is to bridge the semantic gap between the information sources. This consists of mapping the schema and instances of the ontologies used.

Schema matching is a straightforward task in our case. Since the ontologies are known, we can simply insert statements that link corresponding classes in related ontologies. For example, we can state that the Author class of the SWRC ontology is a subclass of the Person class of the FOAF ontology, reflecting the rather trivial knowledge that authors of publications are people.[13]

The matching of instances is a more difficult task and it would be close to impossible to automate without the use of advanced knowledge technology.[14] Identity reasoning is required to establish the identity of objects —in our case individuals— across multiple sources of information, based on the fragments of information available in the various sources. The technical details have been discussed in Chapter 6.

Semantic technology also enables us to reason with social relationships. For example, we have added a rule to our knowledge base which states that the co-authors of publications are persons who know each other. Similarly, the reasoning engine concludes that senders and receivers of emails know each other. In the future, the technology will also allow us to build more refined vocabularies of social relationships, for example to include negative relationships. (The current FOAF ontology only contains a single knows relationship).

9.2.3 Visualization and Analysis

The web interface of Flink allows visitors to browse and visualize the aggregated information about the social connectivity and professional interests of Semantic Web researchers. Researchers can also download their profiles in FOAF format. Although it is not possible to edit the information on site, researchers can take the FOAF files provided and store it at their own sites upon editing it. (The new information will be added at the next update of the website when it is found by the FOAF crawler.) The web interface is built using Java technology, in particular the Java Universal Network Graph (JUNG) API. We encourage the reader to visit the website at `http://flink.semanticweb.org`.

Besides visualization, the user interface also provides mechanisms for computing most of the statistics mentioned in this paper. It is also possible to download the network data and statistics for further analysis in the format used by the Pajek network analysis package [Batagelj and Mrvar, 1998]. Lastly, we provide marker files for XPlanet, an application that visualizes geographic locations and geodesics by mapping them onto surface images of the Earth (see Figure 2.4).

[13] That this may not be the case in the future has been demonstrated recently by a group of MIT students, who have created an application to automatically generate scientific papers. Interestingly enough, their works have been accepted at various conferences as legitimate publications.

[14] Manual solutions to the problem are completely excluded at the scale of our study.

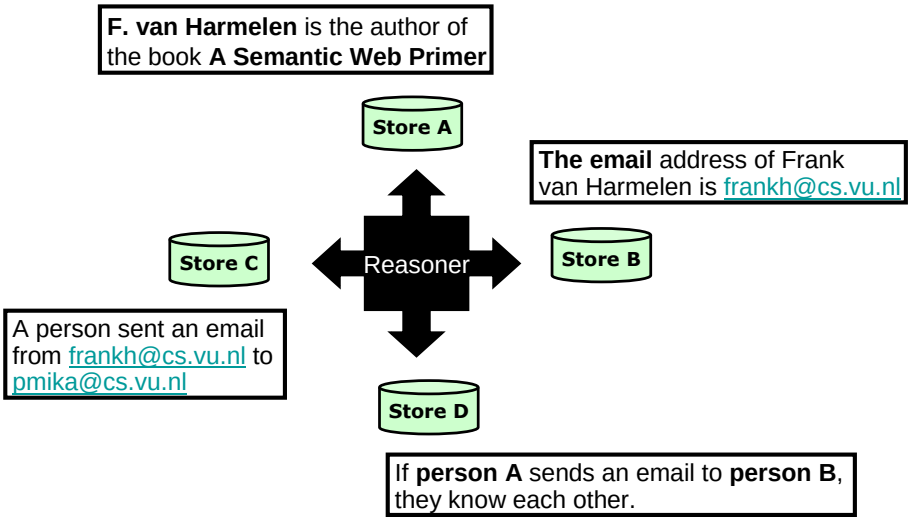


Figure 9.2: An example of identity reasoning. Among others, the reasoner will conclude in this case that Peter Mika knows the author of the book *A Semantic Web Primer* [Antoniou and van Harmelen, 2004]

9.2.4 Electronic data for network analysis

Electronic data is steadily gaining ground as a basis for carrying out network analysis and it is worthwhile to summarize the general benefits and trade-offs associated with the use of electronic data and the specific characteristics of our method.

We begin by noting that the term electronic data covers information originating from electronic documents or systems of any kind, although most studies focus on internet technology, in particular electronic communication networks and online communities. In the first case, the information sources typically include the structure or content of messages passed through email networks, mailing lists, chats, forums, message boards etc. In the case of online communities, the source of information is chiefly the content or linking structure of documents on the World Wide Web or certain parts of it (e.g. blog communities).

The most important benefits of electronic data acquisition can be summarized as follows:

- **Scalability:** Since data extraction does not require the direct involvement of the subjects of the research, the scale of networks that can be studied increases from tens of subjects (typical for interview methods) to hundreds, thousands or more. This also enables longitudinal designs with an arbitrary number of time points, while traditional studies on network evolution typically rely on data collected at only two distinct time points.

- **Unobtrusiveness, lack of bias:** Electronic data collection is unobtrusive and leaves much less possibility for bias than traditional data collection.

- **Repeatability:** As a matter of principle, the same extraction method applied to the same set of data should always produce the same results. This is difficult, if not impossible to achieve with interview or survey techniques of data collection, but an inherent property of automated information extraction methods. This means that research results can be reproduced by others at any time.

While these benefits are common to all methods (including ours), it is important to point out the difficulty in contrasting the related work in this area. Different sources dictate different methods for extracting networks from the underlying data and even different extraction methods applied to the same kind of data can lead to significant differences in the results. As a consequence, researchers report conflicting experiences with regard to the success of using electronic data for network analysis.

This has also increased the caution (if not suspicion) of social scientists, rightfully concerned by the quality of data. As an illustration, we would only like to list some of the possible sources of errors in our own method of mining web-based information sources:

- **Multiplexity**

  One might have noted already that the network obtained from mining the Web is a multiplex network on its own, possibly reflecting the co-authorship network, the discussion networks obtained from emails or some other relationship. A closer look at the results for a single person (Frank van Harmelen) shows that 44 of the first 100 results returned (from a total of about ten thousand) relate to publications and 9 to emails. (Note that the same publication may be referenced in different web pages.) Nevertheless, this network may complement the other networks for different types of relationships (e.g. project co-participation) and data missing from the other sources (e.g. we may not be aware of all mailing lists related to the Semantic Web).

- **Errors in the extraction of specific cases**

  The network is also bound to contain errors due to the method of collection. The search for co-occurrence is carried out on the syntactic level and shows the typical drawbacks of internet search. For example, it is possible that some of the returned pages are about a different person than the one intended by the query. Ambiguity particularly effects people with common names, e.g. Martin Frank. This danger is mitigated by including the disambiguation term in the query.

  Queries for researchers who commonly use different variations of their name (e.g. Jim Hendler vs. James Hendler) or whose names contain international characters (e.g. Jérôme Euzenat) may return only a partial set of all relevant documents known to the search engine.[15] Name ambiguity also effects Google Scholar, which

[15] Worthwhile to note that the ambiguity of web searches with respect to the content is precisely the problem addressed by Semantic Web technology, in particular FOAF for finding people.

is itself based on Information Extraction technology. A typical error is that a person named "York Sure" is identified as a co-author of publications that are published in New York.

With respect to our use case, the situation is analogous to obtaining incorrect data on a network questionnaire for a part of the respondents, namely those with problematic names. However, this will not effect the significance of the results as the fraction of the cases effected this way remains small.

- **General noise**

  Errors in information extraction not only effect specific cases, but also create a general noise. For example, a co-occurrence of names on a web page need not indicate any social relation in the sociological sense and may be in fact a pure coincidence (e.g. names in a phone directory). The strength of support may also be effected by the coverage and reliability of the search engine. Such noise in the data, however, could only result in a Type II error, i.e. missing significant results in the data.

We believe that these difficulties should not be a source of discouragement as it is possible to validate electronic data and reduce error through triangulation. Validation, as carried out for example in the work of Kretschmer and Aguillo [Kretschmer and Aguillo, 2004], consists of establishing the connection between real world networks in science and their online images. Note that such a validation does not negate the benefits of electronic methods, since the collection of manual data ("the golden standard") only needs to be carried out once and even then on a sample of the population. In our current work, we do not compare our network to any golden standard on a dyadic level, but we will show the correspondence between real world status and centrality in the online network.

We also carry out triangulation by relying on multiple sources of information. Again, this approach is made possible by the co-identification of actors in various sources using semantic technology. Although we cannot ensure that our networks precisely map particular kinds of real world networks, this is not required either. Namely, we are only interested in the effect of the online networks on scientific performance, and not the mechanisms through which networks arise. For our purposes, aggregating network information serves the practical use of increasing the already significant portion of the variance in individual performance that we can explain through online network effects.

9.3 Results

We have mapped the structure of the Semantic Web community by combining knowledge from three types of information sources as described in Section 9.2. The actual data set collected on March 17, 2005 contains ties based on 337000 Semantic Web-related web

Pearson	e-mail	pub	web
e-mail	1.000		
pub	0.072	1.000	
web	0.064	0.326	1.000

Table 9.1: Pearson correlations of the three networks extracted from e-mail lists, a publication database (Google Scholar) and web pages.

pages[16], a collection of 13323 messages from five mailing lists[17] and 4016 publications from Google Scholar.

The network extracted from e-mail is slightly more similar to publications than webpages (especially if we raise the threshold for emails), while webpages and publications are much more correlated. This confirms our intuition that webpages reflect publication activity more than the discussion networks of emails. (Table 9.1 shows the results of a QAP analysis performed with UCINET.)

In the following, we take the aggregation of the networks as the object of our study, despite the high correlations between the networks. We do so because all networks contain a number of unique ties beyond the overlap. (For example, email lists reveal working group collaborations that may not be manifested in publications.) The aggregated network thus contains a tie between two persons if there is a tie in either the web, email or publication networks. We are not aggregating the weights from these underlying networks as these weights are measured in different units (association weight, number of emails, number of publications), which are difficult to compare.

9.3.1 Descriptive analysis

Out of the 607 actors, 497 belong to the main component of our network. This connected component itself shows a clear core-periphery structure, supporting our original choice for the boundary definition of the Semantic Web community. (This would not be the case if we would see, for example, two distinct cores emerging.) The single and continuous core/periphery analysis performed with UCINET suggest core sizes of 66 and 114 respectively, where the membership of the larger core subsumes the smaller core with only three exceptions. (The concentration scores show a fairly even decline from their maxima, which suggests that the clusters outside the core are not significant in size and cohesiveness compared to the core.) The presence of a single, densely connected

[16]This count does not take multiplicity into account, i.e. a web page may be tied to more than one name. At the time, there were altogether roughly five million pages on the Web where the term “Semantic Web” is mentioned. In general this shows that the community is highly visible on the Web: in comparison, there were about 13 million pages with the term “Artificial Intelligence” and about 1.2 million pages with the term “social networks”.

[17]These are the rdf-interest, public-swbp-wg, www-webont-wg, public-webont-comments, semantic-web mailing lists, all maintained by the World Wide Web Consortium.

core also means that the measures of coreness, closeness and betweenness are highly correlated in our network.[18]

There is also compelling evidence that measures of the centrality of actors coincide with real-world status in the Semantic Web community. In Figure 9.3, we have listed the top ranking actors according to our centrality measures and labelled them with their positions held in the community. These positions include chairmanship of the ISWC conference series and editorial board membership at the Journal of Web Semantics[19] and the IEEE Intelligent Systems journal[20], the two main sources of Semantic Web-related publications. We also looked at the chairmanship of working groups of the World Wide Web Consortium (W3C), the most influential standards organization in the Semantic Web area.

Ian Horrocks, Dieter Fensel, Frank van Harmelen have been the chairs of the three ISWC conferences held up to date (2002-2004). Stefan Decker and Deborah McGuinness were two of the four chairs of the Semantic Web Working Symposium (SWWS) held in 2001. Rudi Studer and Stefan Decker represent also two of the four editors' in chief of the recently established Journal of Web Semantics. Deborah McGuinness, Frank van Harmelen, Jim Hendler, Jeff Heflin, Ian Horrocks and Guus Schreiber have been co-chairs and/or authors of key documents produced by the Web Ontology (OWL) Working Group of the W3C. Guus Schreiber is also co-chair of Semantic Web Best Practices (SWBP) Working Group, a successor of the OWL group. Jim Hendler is also the current editor-in-chief of the IEEE Intelligent Systems journal. Carole Goble, Toru Ishida and Rudi Studer are joint editors-in-chief of the Journal of Web Semantics.

By looking at the table we can note that all of the common measures of centrality assign high scores to actors with real world status, and we can also ascertain that there are no key position holders of the community whose names would not appear among the first 20 ranks (first three columns). It is also clear that most of the influential members of the community are also successful in terms of the number of publications (fourth column). In terms of impact, i.e. the average number of citations per publication, however, there are members of the community who perform higher than the position holders (fifth column). The explanation is that some peripheral members of the community have highly successful publications in related areas (e.g. agent systems or XML technology). These publications mention the Semantic Web, but are targeted at a different audience than the Semantic Web community.

Despite the overwhelming presence of the core, we can still observe significant clusters outside the core and there is also some remaining clustering within the core. The analysis of overlapping cliques shows that the largest, most cohesive cluster outside the core is formed by researchers working on semantic-based descriptions of Web Services, in particular members of the DAML-S coalition. The recently popular topic of Seman-

[18] In an ideal C/P structure, betweenness correlates highly with closeness, because actors in the core lie on a large portion of the geodesic path connecting peripheral actors. In other words, peripheral actors have to go through actors in the core to reach each other. Similarly, coreness and closeness correlate because actors in the core are close to each other as well as to actors on the periphery, while peripheral actors are only close to actors in the core, but not to each other.

[19] Elsevier, see `http://www.websemanticsjournal.org/`

[20] IEEE Computer Society, see `http://www.computer.org/intelligent/`

Indegree		Closeness		Structural Holes		Publications		Impact	
Name	Value	Name	Value	Name	Value	Name	Value	Name	Value
Steffen Staab	119	Ian Horrocks	0.476	Ian Horrocks	113	Steffen Staab	81	Rakesh Agrawal	684
Dieter Fensel	114	Steffen Staab	0.469	Steffen Staab	105	Dieter Fensel	69	Daniela Florescu	191
Stefan Decker	95	Dieter Fensel	0.468	Dieter Fensel	99	Mark Musen	65	David Kinny	180
Enrico Motta	61	Frank v. Harmelen	0.467	Frank v. Harmelen	91	Ian Horrocks	57	Ora Lassila	166
Frank v. Harmelen	59	Stefan Decker	0.458	Stefan Decker	80	Alexander Maedche	53	Honglei Zeng	153
Raphael Volz	59	Rudi Studer	0.438	Rudi Studer	63	Rudi Studer	50	Stuart Nelson	117
Ian Horrocks	55	Enrico Motta	0.434	Guus Schreiber	48	Amit Sheth	47	Michael Wooldridge	91
Sean Bechhofer	48	Sean Bechhofer	0.427	Enrico Motta	44	Katia Sycara	46	Ramanathan Guha	85
Katia Sycara	48	Carole Goble	0.425	Raphael Volz	43	Frank v. Harmelen	42	Donald Kossmann	83
York Sure	47	Ying Ding	0.424	York Sure	43	Carole Goble	42	Sofia Alexaki	61
Carole Goble	46	Guus Schreiber	0.421	Tim Finin	43	Wolfgang Nejdl	42	Laks Lakshmanan	60
Guus Schreiber	46	York Sure	0.408	Sean Bechhofer	42	Stefan Decker	41	Paolo Atzeni	57
Rudi Studer	46	Peter Crowther	0.407	Katia Sycara	41	Tim Finin	41	Michael Uschold	56
Peter Crowther	40	Alain Leger	0.405	Carole Goble	36	Chen Li	41	Richard Fikes	56
Deborah McGuinness	37	Raphael Volz	0.405	Ora Lassila	27	Enrico Motta	40	Ray Fergerson	55
Ying Ding	35	Herman ter Horst	0.403	Chen Li	26	Nicola Guarino	34	Boris Wolf	53
Jean F. Baget	34	Jim Hendler	0.401	Richard Benjamins	25	John Domingue	33	Michael Lincoln	50
Jim Hendler	33	David Trastour	0.401	Matthias Klusch	24	Gio Wiederhold	30	Fereidoon Sadri	46
Pat Hayes	32	Richard Benjamins	0.400	Michael Sintek	23	Anupam Joshi	30	Yannis Labrou	45

SWWS, ISWC chair (8) **W3C co-chairs (2)** **Journal of Web Semantics (3)** **IEEE Intelligent Systems (2)**

Figure 9.3: Centrality in the social network of researchers reflects real world status

tic Web Services is rather an application of Semantic Web technology as opposed to the more foundational work on ontology languages (RDF, OWL), which are the main shared interest of those in the core. (The clustering could be partly also explained that many of the senior researchers have a background in agent-based systems and have worked together in the past in that area.) To show that this is clearly a topic-based cluster, we have mapped the association of researchers with the concept "DAML-S" against the social network. As Figure 9.4 clearly illustrates, most of these researchers belong to a relatively densely connected subgroup outside the core. (For more information on the method we use to associate researchers with research ideas, please refer to Section 9.2).

9.3.2 Structural and cognitive effects on scientific performance

The social network literature has debated the effect of structure on performance from the perspective of the effects of close interconnectedness versus a sparser network [Burt, 2000]. The basic arguments for the positive effects of a dense interconnected network are that these ties foster trust, identification and these combined lead to an easier exchange of information [Coleman, 1988]. Opposite this argument stands the argument of diversity, by incorporating ties with diverse other groups through the occupation of a structural hole position more new ideas may be encountered and incorporated into one's work [Burt, 2004].

In small scale situations it has been shown that communication ties that bridge a variety of different groups lead to higher performance as did network density [Zuckerman and Reagans, 2001]. In a study of researchers working on the development

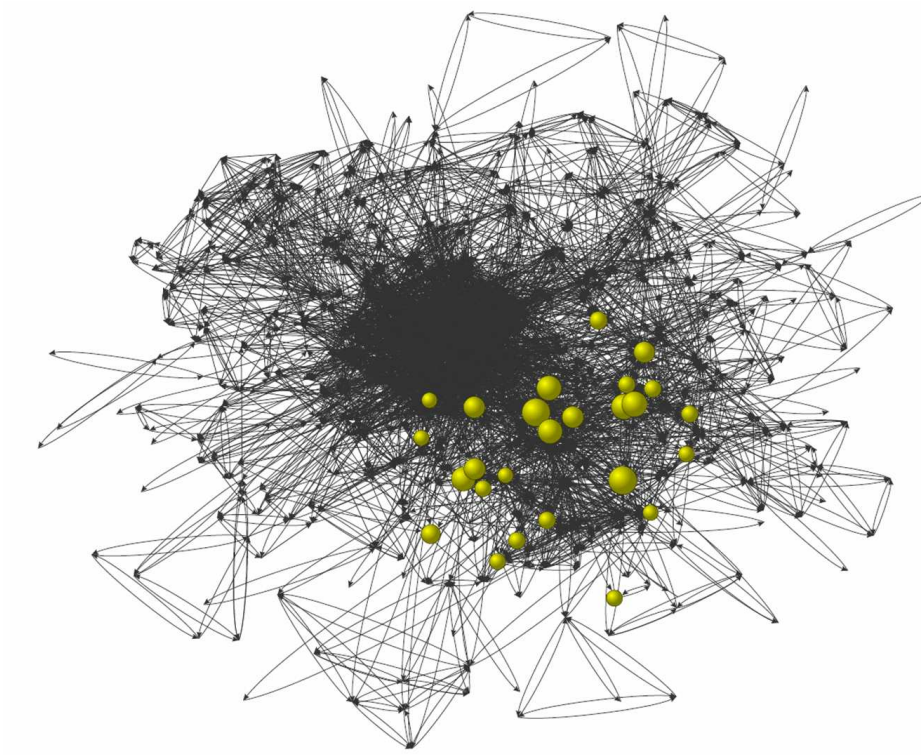


Figure 9.4: Researchers associated with the concept DAML-S form a cluster outside of the core.

of software, centrality measures have been shown to correlate with scientific productivity. An analysis of email messages in this group (about 50 members) showed that centrality correlated strongly with scientific performance [Ahuja et al., 2003]. Centrality was found to be partly, but not completely, a consequence of functional characteristics of the researchers in the field. The reason centrality influences performance is suggested to be a consequence of the benefits a specific individual has from being receiver of a larger amount of (diverse) information.

In the following, we formulate our hypotheses concerning the effects of social networks on scientific performance. First, we test for the effect of ego-network structure, namely the size and density of ego networks. (As previously mentioned, the size of the ego network is also highly correlated with the centrality of the individual.) Second, we look for the additional effects of cognitive diversity in the ego network.

Our primary measure of performance is the number of publications by a given researcher that has received at least a minimum number of citations (dependent variable TOPPUBLI). Based on the general distribution of the number of citations of all publications, we set this minimum at four, excluding 40% of all publications. (The exclusion of minimally cited publications is a commonly used method for filtering out publications

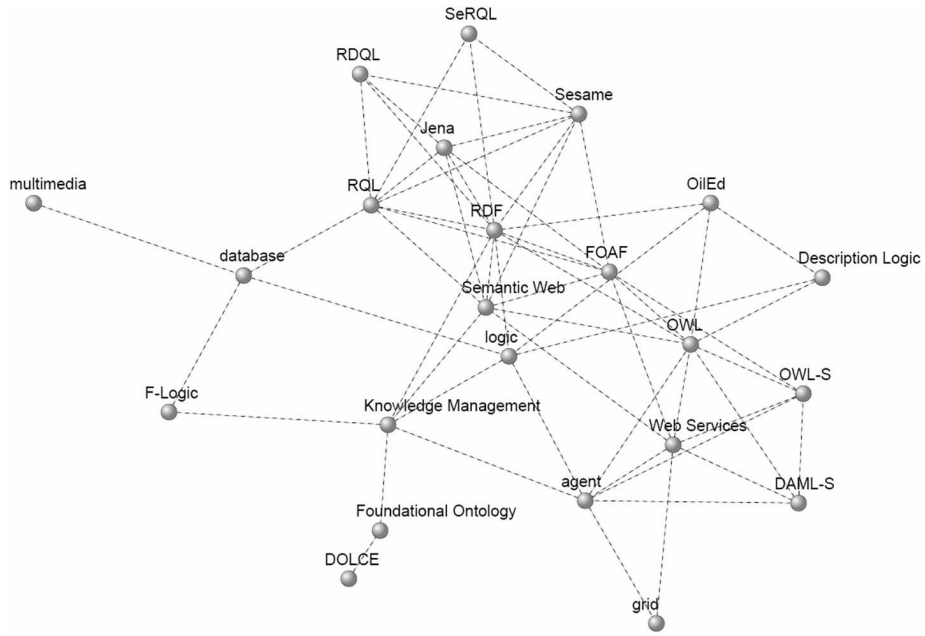


Figure 9.5: The cognitive structure (ontology) of research topics.

that are unlikely to contain innovative results.) Our second, alternative independent variable is the average number of citations per publication (IMPACT). This is a different dimension than the number of publications, because the same impact can be achieved with less or more publications.

We chose the term performance instead of innovativeness when discussing our dependent variables, because publication-based measures offer a very limited way of measuring genuine innovativeness. For example, an article with innovative content may attract few citations if it is published in a journal with limited visibility. On the other hand a survey or review article with minimal innovativeness may attract a large number of citations due to its very nature, especially if published in a high-visibility journal. (These same arguments can also be raised against the usefulness of the impact factor for measuring journal performance.) Nevertheless, publication and citation numbers are still often used in real life to evaluate the performance of researchers.

In reporting our results, we control for experience as we are interested in the unique role of networks in explaining scientific performance. Experience is a personal factor (external to networks) that naturally correlates with both our network variables and our measures of performance.

We measure experience in years of research within the Semantic Web domain, based on the first publication in the area. Therefore, our measure of experience does not in-

Experience	All	$<= 5$	$<= 4$	$<= 3$	$<= 2$
TOPPUBLI	0.665	0.494	0.350	0.299	0.376
sign.	0.000	0.000	0.000	0.000	0.000
df	496	318	248	172	92
IMPACT	0.009	0.286	0.195	0.254	0.269
sign.	0.848	0.000	0.002	0.001	0.009
df	496	318	248	172	92

Table 9.2: DEGREE controlled for NEWEXP

clude (possibly extensive) research experience in other domains, before or in parallel to involvement in Semantic Web research. Further, we do not consider the case of researchers who give up Semantic Web research (or researcher altogether). However, we expect this to be quite rare at this point in time.

The effect of network structure on performance

A closer examination of Ronald Burt's measure of effective size reveals that a structural hole has two distinct components: the size and efficiency of the ego-network [Borgatti, 1997]. In fact, in the simplest case of a network with undirected, unvalued ties, Burt's measure turns out to be equal to the number of direct ties of an actor (degree) minus the clustering coefficient scaled by a factor of $n - 1$, where n is the number of nodes.

In the following, we examine the separate contribution of these components.

Hypothesis 1a. The number of ties to alters is positively related to performance.

Figure 9.2 shows the partial correlation between the number of ties of an individual and our dependent variables, controlling for experience. The first column of the table shows the results for all cases, while the columns to the right show correlations for sections of the populations with less than five, four, three or two years of experience.[21]

In general, we can note that the number of ties explains a significant portion of the publication performance measured in the number of publications. A large social network is thus either the cause or the effect of publishing activity.

In the general population, however, degree is not significantly related to impact. In other words, high impact does not necessarily require a large social network and vice versa. However, it also seems that younger researchers are still able to turn the informational advantages of social access into a higher impact of their publications.

Hypothesis 1b. A dense network of ties among alters (closed network) is negatively related to performance.

[21] If we look at the separate effects of in- and out-degree instead of the full size of the ego network, we find that in-degree is more correlated with the number of publications, while out-degree is more correlated with impact. (Due to limitations of space, we omit the data.) Unreciprocated ties resulting from the analysis of web pages represent a relationship that is more important for the sending actor than the receiving actor.

Experience	All	$<= 5$	$<= 4$	$<= 3$	$<= 2$
TOPPUBLI	-0.146	-0.129	-0.200	-0.179	-0.239
sign.	0.001	0.017	0.001	0.013	0.015
df	525	342	267	190	101
IMPACT	-0.066	-0.080	-0.072	-0.144	-0.205
sign.	0.128	0.140	0.236	0.047	0.037
df	525	342	267	190	101

Table 9.3: CLUSTER controlled for NEWEXP

As expected, clustering in the ego-network of the individual is negatively related to publication performance when measured in the number of publications. A dense network is thus an inefficient network as far as publishing is concerned.

The evidence for the negative effect of clustering on the impact of publications is much weaker. It seems that while clustering negatively impacts the number of publications, it has a much smaller effect on impact. We postulate that publications created in a dense network can still have a relatively high impact within a sub-community of researchers.

The effect of cognitive network structure on performance

Burt's measure of structural holes ignores the actual content that moves through the connection provided. More precisely, Burt assumes that different, unconnected subgroups provide unique knowledge to the broker between them, leading to an advantageous position also in terms of access to knowledge. However, there are more direct ways to establish the link between accessing a diversity of knowledge sources and the performance of the individual.

In a number of previous studies the structural hole argument has been translated to the basic idea of a range of informational sources [Reagans and McEvily, 2003]. The manner in which this variety has been constructed differs in the studies that appeared until now. One example is a study in which organizational variety is taken as a proxy for diversity. Baum et al. considered companies, government agencies and firms with different industrial backgrounds as providing variety [Baum et al., 2000]. In their study on knowledge transfer, Reagans and McEvily used a functional description of roles and variety of expertise [Reagans and McEvily, 2003].

In a relatively homogeneous research field, we expect that cognitive differences may drive the process of innovation. We expect that differences in the research profile of the ego and his alters may benefit the individual in addition to the already proven positive effect of a large and efficient social network.[22]

[22] Note that while we take the cognitive structure as a given, related work by Mutschke and Quan Haase looks at social network-based explanations for the development of the cognitive structure of scientific communities [Mutschke and Haase, 2001]. The authors suggest that the most connected actors (actors with a higher degree centrality) are likely to work on the more central research themes. Renner hypothesizes that the opposite is also true, namely that new ideas are likely to originate from the most peripheral actors. However, such a hypothesis

Hypothesis 2a. Access to cognitive diversity through networks is positively related to performance, especially for younger researchers.

In the following, we measure the cognitive diversity in the ego-network by looking at the difference between the research interests of the ego and his or her alters. We will say that a (structural) tie is a content tie, if there is at least one interest of the alter that is not a current research interest of the ego. We measure diversity by counting the number of content ties of an ego (content-degree). We stipulate positive effects on scientific performance in particular for younger researchers. We believe that senior researchers would be less susceptible to content effects as they can rely on junior researchers in their network (positional advantages) and their functional ties for greater publication performance.

Variable	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
(Constant)	0.736‡	0.883‡	-0.635†	2.452‡	2.946‡	1.106
	(0.000)	(0.000)	(0.041)	(0.000)	(0.000)	(0.265)
DEGREE	0.057‡	-0.036	-0.045	0.144‡	-0.239‡	-0.250‡
	(0.000)	(0.190)	(0.073)	(0.000)	(0.003)	(0.002)
CONTENTD		0.292‡	0.293‡		1.210‡	1.211‡
		(0.000)	(0.000)		(0.000)	(0.000)
NEWEXP			0.737‡			0.893†
			(0.000)			(0.027)
R^2	0.106	0.176	0.316	0.074	0.205	0.228

Figure 9.6: Results of linear regression with the number of publications (TOPPUBLI) as dependent variable (Model 1-3) and the average number of citations (IMPACT) as dependent variable (Model 4-6). N=172, †$p < 0.05$, ‡$p < 0.01$

To show the unique contribution of cognitive diversity towards explaining scientific performance, we perform a linear regression with experience, degree and content-degree as predictors (independent variables) and the number of publications (TOPPUBLI) and average citation (IMPACT) as the outcome (dependent) variables. (We limit our investigations to scientists with at most four years of experience.) The results, shown in Figure 9.6, indicate that the unique contribution of content degree is significant in both cases. (In fact, in the final models the coefficient of degree is not significantly different from zero any more.) We also find evidence that access to cognitive diversity has a particularly large effect on the impact of the publications.

is difficult to prove or refute in practice: the boundary of a scientific network is always fuzzy and a peripheral actor may have many connections to actors outside of the community under investigation.

9.4 Conclusions and Future Work

With our interdisciplinary approach to the study of scientific communities, we are aiming to contribute to both the methods of network analysis and the social theory of research and innovation.

In our methodology, we build on the possibilities offered by Semantic Web technology in the aggregation of the data that we have collected from a number of freely accessible online information sources. The use of freely available electronic data (web pages, publications, mailing lists) not only lowers the cost of studying science communities, but also enables us to significantly increase the scale and longitude of our studies. Further, the reuse of multiple information sources allows us to gain a more complete picture of the community under investigation. Semantic technology is crucial for dealing with the arising heterogeneity.

With respect to our method, we note that it is applicable to a broader range of communities than the one featured in the current study. The few existing comparative studies in webometrics (web-based scientometrics) suggest that real-world networks of largely academic research communities are closely reflected on the Web [Heimeriks et al., 2003, Kretschmer and Aguillo, 2004]. This suggest that our system could be used to generate networks of scientific communities in different areas, potentially on much larger scales. With different sources of data, the framework could also be used to visualize communities in areas other than science, e.g. communities of practice in a corporate setting. As our social lives will become even more accurately traceable through ubiquitous, mobile and wearable computers, the opportunities for social science based on electronic data will only become more prominent.

In the above, we have shown the immediate benefits of our methodology by applying it toward a network study of the Semantic Web community. Based on our data set, we have proved the positive effects of a large, efficient (sparse) network on the innovativeness of researchers, confirming the benefits attributed to Structural Holes [Burt, 2004]. We have extended the well-known structural analysis of this scientific community with a novel analysis of the content of relationships. We have shown that diverse cognitive networks have a positive impact on performance beyond the structural effects. Our measure of content degree results to be a much better predictor than the conventional measure of degree for both the number of publications and the average number of citations.

We are planning to extend our work in this direction, e.g. by investigating whether cognitive diversity in the ego network could have a negative effect in cases where the distance between research areas is overly large. We are also developing measures to study the expected positive effect of achieving a diverse cognitive network with a minimal investment in social ties. Planned improvements to our information retrieval methods should also enable us in the future to determine more precisely the interests of individual researchers in the community. Lastly, we are tracking the development of the Semantic Web community over time using our electronic methods of data collection, providing a wealth of data for future work.

Chapter 10

Ontologies are us: emergent semantics in folksonomy systems

According to the most cited definition of the Semantic Web literature, an ontology is an explicit specification of the conceptualization of a domain [Gruber, 1993]. Guarino clarifies Gruber's definition by adding that the AI usage of the term refers to "an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words" [Guarino, 1998]. An ontology is thus engineered by —but often for— members of a domain by explicating a reality as a set of agreed upon terms and logically-founded constraints on their use.

Conceiving ontologies as engineering artifacts allows us to objectify them, separate them from their original social context of creation and transfer them across the domain. Problems arise with this simplistic view, however, if we consider the temporal extent of knowledge. As the original community evolves through members leaving and entering or changing their commitments, a new consensus may shape up, invalidating the knowledge codified in the ontology.

To address the problem of ontology drift, several authors have suggested *emergent semantics* as a solution [Aberer et al., 2004]. The expectation is that the individual interactions of a large number of rational agents would lead to global effects that could be observed as semantics. Ontologies would thus become an emergent effect of the system as opposed to a fixed, limited contract of the majority. While the idea quickly caught on due to the promise of a more scalable and easily maintainable Semantic Web, the agreement so far only extends to the basic conditions under which emergence would take place. The vision is a community of self-organizing, autonomous, networked and localized agents co-operating in dynamic, open environments, each organizing knowledge (e.g. document instances) according to a self-established ontology, establishing connections and negotiating meaning only when it becomes necessary for co-operation.

Beyond the reasonable belief that individual actions in such a semantic-social network would lead to ontology emergence, there is a lack of an abstract, empirically verifiable model of such a system that could also explain the process of emergence. Thus there appears to be a large conceptual gap in the literature between the vision and the details of implementations of various semantic architectures based on P2P, Grid, MAS and web technology.

In this Chapter, we take a step back and formulate a generic, abstract model of semantic-social networks (Section 10.1), which we will call the Actor-Concept-Instance model of ontologies. This model is built on an implicit (albeit crucial) realization of emergent semantics, namely that meaning is necessarily dependent on a community of agents. Inspired by social tagging mechanisms, we represent semantic-social networks in the form of a tripartite graph of person, concept and instance associations, extending the traditional concept of ontologies (concepts and instances) with the social dimension. We will show how lightweight ontologies of concepts and social networks of persons emerge from this model through simple graph transformations. In Section 10.2 we will demonstrate these effects based on two independent, large scale data sets. In Section 10.3, we evaluate one of our emergent ontologies (the result of a social-network based ontology extraction process) against the results of the traditional method of ontology extraction based on co-occurrence. Lastly, we conclude with a discussion of future work in Section 10.4.

10.1 A tripartite model of ontologies

While expert systems designed for centralized, controlled environments benefit greatly from the increasing expressivity of ontology languages such as OWL, especially in domains that lend themselves naturally to formalization such as engineering and medicine, lightweight ontologies expressed in RDF(S) have spread and caught on in the loosely controlled, distributed environment of the Web [Mika and Akkermans, 2004].

The tendency towards lightweight, easily accessible mechanisms for ontology and metadata creation is best evidenced by the recent appearance of folksonomies. Folksonomy (from folk and taxonomy) is a neologism for a practice of collaborative categorization using freely chosen keywords.[1] Folksonomies (also called social tagging mechanisms) have been implemented in a number of online knowledge sharing environments since the idea was first adopted by the social bookmarking site del.icio.us in 2004.

The idea of a folksonomy is to allow the users to describe a set of shared objects with a set of keywords of their own choice. What the objects are depends on the goal of the site: while bookmarks are the object of classification in del.icio.us, photos are shared in Flickr, scientific publications are tagged in CiteULike, while 43Things allows users to

[1] "A portmanteau of the words folk (or folks) and taxonomy, the term folksonomy has been attributed to Thomas Vander Wal. Taxonomy is from 'taxis' and 'nomos' (from Greek). Taxis means classification. Nomos (or nomia) means management. Folk is people. So folksonomy means people's classification management." Source: Wikipedia.

share their goals and plans (e.g. to travel or loose weight) by annotating their descriptions with keywords and connecting users with similar pursuits.[2]

It is important to note that in terms of knowledge representation, the set of these keywords cannot even be considered as vocabularies, the simplest possible form of an ontology on the continuous scale of Smith and Welty [Smith and Welty, 2001]. First, the set of words is not fixed. In fact, the users form no explicit agreement at all about the use of words, not even in the form of incremental, need-based, local and temporary agreements suggested by the research on emergent semantics [Aberer et al., 2004]. Yet, the basic conditions of emergent semantics are given and as we will show there is semantics emerging at the scale of these systems. Second, although we use the term concept in the following, it is clear that there is no one-to-one correspondence between concepts and keywords. It is not always possible for the users to express a complex concept with a single keyword and thus they may use more than one tag to express the concept association that the item brings up in them. Lastly, the instances of folksonomies are instances only in the sense of classification.

In order to model networks of folksonomies at an abstract level, we will represent such a system as a tripartite graph with hyperedges. The set of vertices is partitioned into the three (possibly empty) disjoint sets $A = \{a_1, \ldots, a_k\}$, $C = \{c_1, \ldots, c_l\}$, $I = \{i_1, \ldots, i_m\}$ corresponding the set of actors (users), the set of concepts (tags, keywords) and the set of objects annotated (bookmarks, photos etc.) In effect, we extend the traditional bipartite model of ontologies (concepts and instances) by incorporating actors in the model.

In a social tagging system, users tag objects with concepts, creating ternary associations between the user, the concept and the object. Thus the folksonomy is defined by a set of annotations $T \subseteq A \times C \times I$. Such a network is most naturally represented as hypergraph with ternary edges, where each edge represents the fact that a given actor associates a certain instance with a certain concept. In particular, we define the representing hypergraph of a folksonomy T as a (simple) tripartite hypergraph $H(T) = \langle V, E \rangle$ where $V = A \cup C \cup I$, $E = \big\{\{a, c, i\} \mid (a, c, i) \in T\big\}$.

Tripartite graphs and hyperedges are rather cumbersome to work with. However, we can reduce such a hypergraph into three bipartite graphs (also called two-mode graphs) with regular edges. These three graphs model the associations between actors and concepts (graph AC), concepts and objects (graph CO) and actors and instances (graph AI). For example, the AC valued bipartite graph is defined as follows:

$AC = \langle A \times C, E_{ac} \rangle,\ E_{ac} = \{(a, c) \mid \exists i \in I : (a, c, i) \in E\},\ w : E \to \mathbb{N},\ \forall e = (a, c) \in E_{ac},\ w(e) := |\{i : (a, c, i) \in E)\}|$

In words, the bipartite graph AC links the persons to the concepts that they have used for tagging at least one object. Each link is weighted by the number of times the person has used that concept as a tag. This kind of graph is known in the social network analysis literature as an affiliation network [Wasserman et al., 1994], linking people to affiliations with weights corresponding to the strength of the affiliation. An affiliation network can be used to generate two simple, weighted graphs (one-mode networks) showing the sim-

[2] `http://del.icio.us`, `http://www.flickr.com`, `http://www.citeulike.org`, `http://www.43things.com`

ilarities between actors and events, respectively. (At this point it is recommended to dichotomize the graph by applying some threshold.)

The process of folding a bipartite graph (the extraction of a one-mode network) can be most easily understood by looking at the matrix form of the graph. Let's denote this matrix as $\mathbf{B} = \{b_{ij}\}$. As discussed before, $b_{ij} = 1$ if actor a_i is affiliated with concept c_j. We define a new matrix $\mathbf{S} = \{s_{ij}\}$, where $s_{ij} = \sum_{x=1}^{k} b_{ix}b_{xj}$. In matrix notation $\mathbf{S} = \mathbf{B}\mathbf{B}'$. This matrix, known as the co-affiliation matrix, defines a social network that connects people based on shared affiliations. In our case the links are between people who have used the same concepts with weights showing the number of concepts they have used in common. The dual matrix, $\mathbf{O} = \mathbf{B}'\mathbf{B}$ is a similar graph showing the association of concepts, weighted by the number of people who have used both concepts as tags. Note that in both graphs the diagonal of the corresponding matrices contains the counts of how many concepts or persons a given person or concept was affiliated with in the bipartite graph. We can use these values to normalize the association weights (e.g. by calculating the Jaccard-coefficient) and then filtering again based on the relative weights. In case of the $\mathbf{S}$ social network, for example, this means that we have taken into account the relative importance of the link between persons.

In summary, the AC graph, the affiliation network of people and concepts can be folded into two graphs: a social network of users based on overlapping sets of objects and a lightweight ontology of concepts based on overlapping sets of communities. Thus in this simple model, social networks and semantics are just flip-sides of the same coin: the original bipartite graph contains all the information to generate these networks, while it is not possible to re-generate the original graph from them.

The other two bipartite graphs that we derived from the original tripartite model can also be folded into one-mode networks in a similar fashion. In particular, the CI graph leads to another semantic network, where the links between terms are weighted by the number of instances that are tagged with both terms. This type of semantic network is of a much more familiar kind: it mimics the basic method applied in text mining, where terms are commonly associated by their co-occurrence in documents. The AI graph results in another social network of persons, where the weight of a pair is given by the number of items they have both tagged. We also get a network of instances, with associations showing the number of people who have tagged a given pair of instances.

In the following we focus our attention to the two lightweight ontologies based on overlapping communities (O_{ac}) and overlapping sets of instances (O_{ci}).[3] The analysis of the emergent social networks is outside the scope of the current Chapter.

10.1.1 Ontology enrichment

The community-based lightweight ontology O_{ac} that we extract from the affiliation network is rather peculiar from a knowledge representation perspective. Unlike the manually constructed thesauri known in the Semantic Web literature (such as WordNet [Fellbaum, 1998]), it more closely resembles the association thesauri studied in

[3]Recall that $O_{ac} = \mathbf{B}'\mathbf{B}$, where $\mathbf{B} = \{b_{ij}\}$ with $b_{ij} = 1$ if actor a_i is affiliated with concept c_j; and $O_{ci} = \mathbf{D}'\mathbf{D}$, where $\mathbf{D} = \{d_{pq}\}$ with $d_{pq} = 1$ if concept c_p is used to tag the instance i_q.

linguistics. An example is the Edinburgh Associative Thesaurus (EAT)[4], which was collected in 1973 via an experiment using a group of university students as subjects [Kiss et al., 1973]. The experiment consisted of handing a list of words to students who were instructed to write down against each stimulus word the first word it made them think of, working as quickly as possible. The obtained words were used in a next round of the experiment. (The cycle was repeated three times, by then the number of different responses was so large that they could not all be re-used as stimuli.)

Our associative ontology is similar to the EAT in that the weights of the links between terms are expressed as the number of people who make that association. The difference is that in the EAT collection, people are prompted explicitly to create links between concepts, while we deduce such links by observing tagging behavior. More importantly, however, both methods have the crucial property that the result clearly depends on the community of people who take part in experiment. The method of ontology engineering is particularly revealing, because once the initial set of words is selected there is only one parameter to the process: the population chosen. (In particular, the knowledge engineer has no other role than handing out questionnaires and collecting the responses.) Some of the results are likely to hold for other communities (like the overwhelming reaction of saying *Noah* when hearing the word *ark*), but many of the aggregated associations are driven by the collective mind set of the subjects of the experiment. A collective mindset that is likely shaped by the well-known law of community formation: interaction creates similarity, while similarity creates interaction.

We can not only repeat the experiments with different communities, but given some information about the social structure of the community, we could also extract local ontologies by limiting our tripartite ontology to the associations of a certain sub-community of actors. Note that this is the principle of locality in action, one of the expected hallmarks of emergent semantics [Aberer et al., 2004]. We will demonstrate this effect in Section 7.2 where we extract an ontology of research topics in the Semantic Web domain.

In modern terms, the EAT is an emergent ontology based on empirical data. Unlike ontologies that are meant to codify fixed agreements, all graphs that we derive are also emergent in the sense of evolving dynamically with the Actor-Concept-Instance network. Changes in the original network can occur in a number of ways. Users may join or leave the community, changing the set of actors. The focus of the community may shift, affecting the set of items tagged and the concepts used. Last, the understanding and use of terms may change, reflecting in the set of associations between concepts and instances created by the users.

Although our association networks are very simple ontological structures, there are several opportunities of enriching them with additional semantics. We start by observing that a significant drawback of the EAT is the heterogeneity of terms. Our emergent ontologies are also likely to contain a diverse mixture of specific and generic terms, i.e. terms that we can unambiguously place in a clearly defined context (e.g. instances such as *Peter*) and terms that can occur in multiple contexts of use (e.g. *war*). From a network view, general words are therefore more likely to bridge different clusters of words,

[4]Consult the EAT online at `http://www.eat.rl.ac.uk/`

while specific terms are expected to exhibit a dense clustering in their neighborhood. This suggest an opportunity to distinguish between these two categories by computing the *clustering coefficient*, the *(local) betweenness centrality* or the *network constraint* on our terms (see Chapter 3). These well-known ego-network measures of Social Network Analysis are readily available in popular network analysis packages such as Pajek [Batagelj and Mrvar, 1998] and UCINET [Borgatti et al., 2002]. Based on the same observation, we also expect that clustering algorithms can help us in finding synonym sets of the more specific terms. There is a wide range of clustering algorithms available in the above mentioned network analysis packages, based on different definitions of cohesiveness.

We may also extract broader/narrower term relations typical of thesauri using set theory. In an ideal situation, we would say that Concept A is a super-concept of Concept B if the set of entities (persons or items) classified under B is a subset of the entities under A ($B \subseteq A \leftrightharpoons A \cap B = B$). We might also add the criterion that the set of A should be significantly larger then the set of B, i.e. $|B|/|A| < k$ for some value of k. In principle, such an ordering allows us to define a Galois lattice using the subset relation. In practice, such a lattice would be very sparse (considering the number of entities and the number of possible subsets over them), so we will approximate this method by looking for near-perfect overlaps, i.e. $|A \cap B|/|B| < n$ for some value of n. Finding appropriate values for the k, n parameters of the model is the task of the researcher.

The reader should note that the meaning of these broader/narrower relations are very different, depending on whether we analyze the O_{ci} or the O_{ac} ontology. In the first case, the interpretation is that all (or most) of the items classified under the narrower term also appear under the broader term. In other words, what we extract is a classification hierarchy. In the second case, the meaning is that all the persons associated with the narrower term are also associated with the broader term. In other words, we extract a hierarchy based on sub-community relationships.

10.2 Case studies

In the following, we demonstrate the broad applicability of the Actor-Concept-Instance model of ontologies by looking at two different semantic social networks. Our first data set comes from an existing web-based social bookmarking tool called del.icio.us (Section 10.2.1), while the second case is built on synthetic data obtained by using web mining techniques (Section 10.2.2). We will show how the abstract model applies to the particular cases and demonstrate our method of ontology emergence based on the graph transformation described above.

10.2.1 Ontology emergence in del.icio.us

According to the definition of author Joshua Schachter, del.icio.us is a social bookmarking tool.[5] Much like the similar functions of browsers, del.icio.us allows users to manage a personal collection of links to web sites and describe those links with one or more

[5] See `http://del.icio.us`

keywords. Unlike stand-alone tools, del.icio.us is a web-based system that allows users to share bookmarks with each other. Bookmarks can be browsed by user, by keywords (tags) or by a combination of both criteria. Further, the user interface encourages exchange by showing how bookmarks are linked together via users and tags. In terms of the Actor-Concept-Instance model, registered users of del.icio.us are the actors who create or remove associations between terms and webpages (instances) by adding or deleting bookmarks.

From the perspective of studying emergence, del.icio.us is remarkable for the dynamics of its user base. The young, technologically aware community gathering around the site closely follows the latest news and trends in web technology as well as the evolving vocabulary of the field. Beyond technology, del.icio.us users also post bookmarks related to current topics in politics, media, business and entertainment. The emphasis on timeliness is reinforced by listing bookmarks in a backward-chronological order as it is typical for blogs.

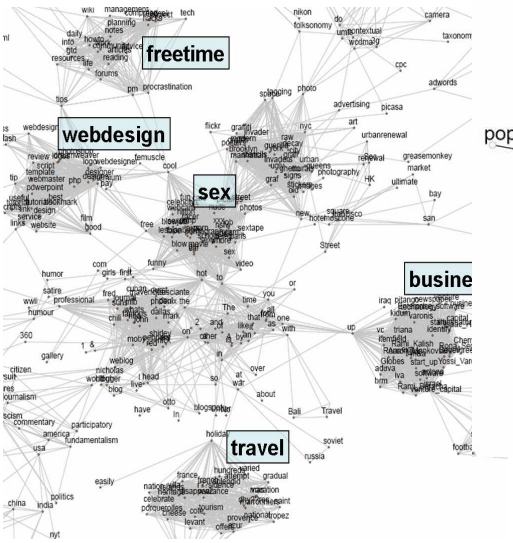


Figure 10.1: The del.icio.us tags associated through co-occurrence on items and the clusters emerging.

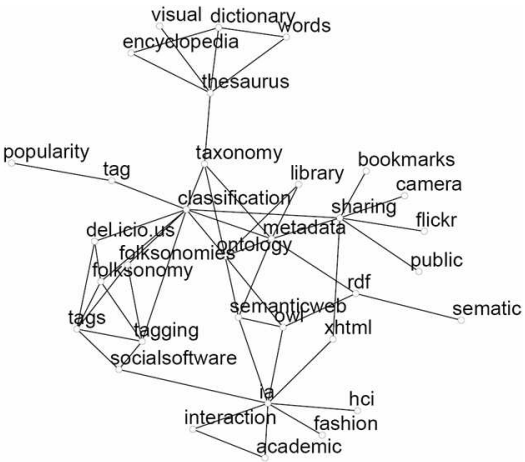


Figure 10.2: Detail view of the del.icio.us tags associated through users: a 3-neighborhood of the term ontology. Note that the term *sematic* is correctly associated, despite the obvious typo.

The process of annotation is made as easy as possible. A single textbox allows users to enter a set of words without any recommendations made by the system. On the downside, this means that synonyms are common in the folksonomy, e.g. "semanticweb", "semweb" are different keywords. Ambiguity is also present, since users often pick overly general terms to describe items (such as "web", "tool" and other popular terms). Further, users often make the mistake to enter key phrases instead of keywords (e.g. "Bill Clinton"), where the words are subsequently parsed as separate tags ("Bill" and "Clinton"); or they escape the one-word-only limitation by concatenating words. Case

sensitivity and the use of punctuation marks further pollute the del.icio.us namespace. However, at the scale of system (over 30 thousand registered users in December, 2004) the imperfections of tagging are reduced to an acceptable level. On the plus side, users benefit from instant gratification in the form of linkage to other relevant, timely, socially-ranked posts.

del.icio.us exposes tagging data in the form of RSS feeds, which we have collected using a focused RDF crawler. The crawler was initialized with the single most popular tag ("web") and have traversed the RSS network in a breadth-first-search manner, following links to tags mentioned in the descriptions of items. The sample data that we collected —over a million triples of RDF— was stored using the Sesame storage and query facility [Broekstra et al., 2002]. The sample represents 51852 unique annotations of 30790 URLs, by 10198 persons using 29476 unique tags.[6]

Next, we have generated both the Actor-Concept and Concept-Instance graphs. In order to scale down the data set (without loosing much information) and to avoid strong associations with a low support we have filtered out those entities that had only a minimal number of connections, i.e. those tags that had less than ten items classified under them and those persons who have used less than five concepts.

Subsequently, we have extracted the above mentioned two kinds of ontologies by folding these graphs using the network analysis package Pajek. As a reminder, the first ontology (O_{ac}) is based on actors sharing concepts as interests, i.e. the associations reflect overlapping communities of interests, while the second network (O_{ci}) reflects the co-occurrence of tags on items. We have filtered the networks based on the absolute strength of associations. Next, we applied geometric normalization to the resulting graphs and filtered edges again based on the relative strength of the associations. We have chosen the thresholds in such a way that we obtain networks of equal size (438 concepts). Figure 10.1 shows a high level view of the O_{ci} graph, Figure 10.2 shows a detailed view of the O_{ac} graph.

The results show clear evidence of emerging semantics in both cases, but the networks we obtain still show very different pictures. With an equal number of vertices, the densities of the two networks are quite different (0.01 for the O_{ci} network, 0.006 for the O_{ac} network), and so is the amount of clustering present (the average clustering coefficients are 0.2 and 0.03, respectively).

The selection of concepts in the two networks is also very different: only 64 concepts are present in both networks of the total of 438 nodes in each graph. (A sample is included in Table 10.1.) A closer look reveals that the concepts within the clusters of the first network are often very specialized terms, while those in between the clusters are overly general terms. A look at the terms with the lowest clustering and highest betweenness centrality confirms this hypothesis. The top five terms with highest betweenness are *up*, *cool*, *hot*, *in*, *to*. Noticeable also is that the terms with the highest clustering and lowest network constraint are those related to sex. As mentioned before, the second network shows much less clustering: overly general and overly specific terms are both missing.

[6]This is a sample of the complete data set because the RSS feeds expose only the latest thirty items for each tag. Further, we stopped crawling after reaching this size. To our knowledge this is still the largest ontology annotation data set ever studied.

O_{ci}	*/GoogleHacks, _0, 04, 1, 2, 2005, 3g, a, A, a9, Aaron_Mankovski, actona, actors, adult, aduva, advice, ajax, all, Allegrini, america, an, and, angeles, apparel, Apple, as, assembly, attempt, attention, attention.xml, aviv, axml, azur
O_{ac}	.net, 3d, 43folders, academic, accessibility, acronym, actionscript, activism, ad, ads, adsense, advertising, advice, advisories, adwords, agile, ajax, amazon, america, analysis, and, Apache, apache, api, app, apple, application, architecture, archive, Art, art, articles, asia, astronomy, atlas, Audio

Table 10.1: Terms starting with "A" or "a" in the two lightweight ontologies generated from the del.icio.us network.

travel	cote, provence, villa, azur, mas, holiday, vacation, tourism, france, heritage
business	venture_capital, enterprise, up, start, venture, newspaper, capital, Segev, pitango, vc
free time	procrastination, info, advice, gtd, life, notes, planning, daily, reading, forums
sex	hot, to, street, pictures, on, photos, free, celeb, adult, lesbian
web design	design, designer, webdesign, premium, logo, logos, dreamweaver, templates, best, good

Table 10.2: The five main clusters of interest based on the Concept-Object network.

The clue to the different qualities of these networks lies in the difference in the way associations are created between the concepts. In the first case, there exists a strong association between concepts if they share a large percentage of items, *independent of the number of users interested in them and regardless whether these associations were added by the same users or not*. The resulting distribution of association weights shows a very slow decline, the average weight is fairly high. In the second case, there is a strong association in the network if two concepts share a large fraction of the users among them, *independent of the number of instances associated with them and regardless whether these terms were added to the same instances or not*. The resulting weight distribution shows a very steep decline, the average weight is fairly low.

This suggests that the first network (O_{ci}) is more appropriate for concept mining. In fact, a λ-set analysis performed with UCINET on a slightly larger network of 751 concepts resulted in meaningful clusters of specific terms, representing various domains of interest in the del.icio.us community. At a level of $\lambda = 20$, we found 5 cohesive groups of concepts that we identified as interests related to travel, business, free time, sex and web design (see Figure 10.2 and Table 10.2).

However, the O_{ci} semantic network ignores the relevance of the individual concepts from the user perspective and as such it gives an inaccurate picture of the community.

Broader	Narrower
rss	atom
cmyk	rgb
cell	umts, wcdma, ev-do
phone	cell
ajax	json
xml	xslt
rdf	owl
flickr	gmail, picasa
ruby	rails
mac	iphoto
java	j2ee
google	gds
search	a9, engine
linux	ubuntu, gnome
flash	actionscript
flickr	lickr, photoset
javascript	xmlhttprequest, dom, sarissa

Table 10.3: Broader/narrower term relations in the technology domain, based on sub-communities in del.icio.us.

Concepts related to sex, for example, get a misleadingly high centrality in the network due to the specificity and extent of the vocabulary used to describe sex-related sites. On the other hand, the more evenly distributed community-based network (O_{ac}) contains concepts that are actually important to del.icio.us users. These concepts almost all come from the computer domain, the apparent core interest of users. The strength of links between the concepts are also a more accurate representation of reality as they are not biased by the actual number of items that have been tagged with them.

The ignorance of the item-based extraction method towards the number of users also makes it problematic to extract taxonomic relations. Namely, many of the relations we extracted are based on the word usage of a small number of users, and in the worst case a single user. The Concept-Actor ontology yields much more easily interpretable results, shown in Figure 10.3. As discussed before, these are sub-community relations: the community associated with a narrower term is a sub-community of the community associated with the broader term. Nevertheless, even here we find an association created by a single story marked by a large number of users. This suggests an improvement to our original method, namely filtering out concepts that have only a limited number of items or persons associated to them. We take this into account as we move on to generalize our method to community-based ontology extraction from Web pages.

We conclude by noting the potential application of the results to improving del.icio.us itself, e.g. by offering search and navigation based on broader/narrower terms. Considering the dynamics of the community and the extent of neologism, the ontologies emerging

from folksonomies such as del.icio.us also have a large potential for enriching established, but slowly evolving linguistic ontologies such as WordNet [Fellbaum, 1998].

10.2.2 Community-based ontology extraction from Web pages

Folksonomies such as del.icio.us are effective, because they attract sizeable subcommunities of users pursuing similar interests. Nevertheless, the community of del.icio.us is still a niche compared to the general web population, just as the number of web sites tagged is only a fraction of the number of pages on the Web.

We would like to show in the following that even without explicitly assigned tags, it is possible to extend the idea of community-based ontology extraction to the Web. Let's suppose that we have a selected a community, whose members will play the role of Actors in our model, and we have prepared a list of terms whose associations we are interested in. The instances of our model are the pages of the Web. Further, we assume that a web page is tagged by a concept if the concept occurs on the page.

Based on these assumptions, the Concept-Instance ontology is straightforward to create: we can use a search engine to obtain page counts for all pairs of concepts and then normalize by their separate page counts. This is the basic co-occurrence analysis method of text mining.

Generating the Actor-Concept ontology requires another broad assumption. We will say that there is an association between a concept, a person and a web page if the name of the person and the label of the concept co-occur on the page. This association represents a weaker commitment than in the case of folksonomies, because it is not guaranteed that the association is made *by* the person. Nonetheless, we can now generate the bipartite graph of persons and concepts by measuring the association using page counts from the search engine.

First, we measure the association between a person (e.g. *"Peter Mika"*) and a concept (e.g. *"Semantic Web"*) by submitting a boolean query combining the two terms (e.g. *"Peter Mika" AND "Semantic Web"*). We normalize the result with the number of pages where the concept occurs. We then repeat this with the same concept and the names of all other members of the target community. We calculate the mean strength of association with the concept of *"Semantic Web"*. Lastly, we associate those members of the community with this concept whose association strength is at least one standard deviation higher than the mean. (Note that this is a slightly more sophisticated method of filtering than a general threshold.) We can now fold the bipartite graph of actors and concepts to obtain the O_{ac} ontology.

Our method of community-based ontology extraction have been implemented as part of the Flink system (see Section 7.2). The system is a web-based presentation of the social networks and research interests of Semantic Web researchers. The community of researchers represented in Flink includes all authors, program committee members and organizers of all past international Semantic Web events from 2001, altogether 607 persons. The system extracts the social network of researchers as described in [Mika, 2004] and associates them with research topics using the search engine Google. [7]

[7] Note that the page counts returned by Google violate some of the assumptions of our set theoretic interpretation of boolean logic. For example, OR queries in Google typically return less items than the terms

Flink can also be used to perform co-occurrence analysis and generate the O_{ci} ontology. We improve the basic method by adding the disambiguation term *"Semantic Web" OR ontology* to the queries sent to the search engine, limiting the items returned to those relating to the Semantic Web.

The resulting ontological structures are not included here due to limitations of space, but we strongly encourage the reader to consult them online[8]. To make the networks comparable, we have included only the 100 strongest associations in each network. Again, we see a significant difference in the set of concepts remaining in the networks. Namely, from the original 60 terms (selected manually from the proceedings of the ISWC events), the method of text mining found the strongest associations between more general terms. Specific concepts related to the Semantic Web seem to float to the periphery and are misplaced in general. For example, the term *FOAF* is related to *XML* and *OWL-S*, technologies not directly related to FOAF. *Annotation* is related to *alignment* and *databases*. The term *ontology* is associated, among others, with *HTML*, *XML* and *databases*, concepts not directly related to the understanding of ontologies in the Semantic Web community.

The O_{ac} association network represents a clear improvement in these respects. The method found correct associations between domain specific concepts. For example, the term *FOAF* is linked here to *Redland* and *Sesame*, the triple stores preferred by FOAF developers for their scalability. Terms related to ontology languages (*OWL*, *RDF*, *OIL*, *DAML+OIL*, *ontology languages* etc.) are correctly clustered together, just as the technologies related to ontology storage (query languages, triple stores), with terms related to ontology development (*OilEd*, *OntoEdit*, *ontology development*) connecting the two clusters. More general technologies are also placed correctly in context, i.e. corresponding to the way they are used in the Semantic Web. For example, *NLP* is tied to the notions of *annotation* and *ontology learning*.

The difference in the node sets can be explained in a similar way as in the case of del.icio.us: the O_{ci} network ignores the overall relevance of these concepts to the Semantic Web community. Considering the associations, we believe that there is another effect in play. By querying the associations of persons first and then linking concepts through overlapping communities, we simulate the effect of first asking the members of the community to associate themselves with certain research interests and then relating these interests through overlapping communities. Overlapping communities turn out to be a stronger link than overlapping sets of web pages. A possible explanation is that even after including the disambiguating term in the query, the search engine still suffers from *knowing too much*, blurring away community-specific interpretations.

10.3 Evaluation

In absence of a golden standard, evaluating the results of ontology learning or ontology mapping is a difficult task: inevitably, it requires consulting the community or communities whose conceptualizations are being learned or mapped. In order to evaluate our

separately, i.e it is possible that $|"A" \wedge "B"| < |"A"|$. Further, the boolean operators are not symmetric: $|"A" \wedge "B"| \neq |"B" \wedge "A"|$.

[8] `http://www.cs.vu.nl/~pmika/research/iswc2005/`

results, we have thus approached in email 61 researchers active in the Semantic Web domain, most of whom are members of the ISWC community and many of them are in the graph-theoretical core of the community[9]. The single question we asked was *In terms of the associations between the concepts, which ontology of Semantic Web related concepts do you consider more accurate?* Lacking a yardstick, there is no principled correct answer to this question that we expected to receive. Instead, we were interested to find out if there is a majority opinion emerging as an answer and if yes, which of the two ontologies (produced by the two different methods) would that majority accept as more accurate.

Many respondents expressed difficulty in answering the question due to the (intentional) lack of further explanations or instructions, e.g. what the associations mean, but also due to the very different node sets of the two semantic networks. Nonetheless, out of the 33 respondents only three persons were not willing to express any preference (even if a slight one) for one network or the other. 23 respondents were members of the ISWC community and 15 of them belong to the core of the community.

The distribution of the answers for the various subgroups are summarized in Table 10.4. First, taking all responses into account, we can conclude that the participants consider the O_{ac} network as a more accurate representation of associations between the concepts than the O_{ci} network (the result is significant at a level of $p = 0.01$). The majority vote becomes even stronger if we consider only the members of the ISWC community, i.e. the persons whose name has been used to extract the semantic network. Thus as a second finding we can also conclude that the O_{ac} network is considered more accurate particularly by those whose names were used in the extraction process. The results become even more conclusive if we only consider the votes from the core members of the community. Based on this finding and assuming a continuum, we can state that the O_{ac} network better reflects the conceptualizations of those closer to the core of the community. Combined together, our findings confirm that the O_{ac} network better reflects the conceptualizations of those involved in Semantic Web research, and this holds especially for those most actively involved in Semantic Web research.

	N	O_{ac}	O_{ci}	Ratio	Sign.	
All	30	22	8	73.3%	0.0055	
ISWC	23	18	5	78.3%	0.0040	
ISWC-core	15	13	2	86.7%	0.0032	

Table 10.4: Results for the comparison of the community-based (O_{ac}) and item-based (O_{ci}) ontology extraction methods.

10.4 Conclusions and Future Work

The Semantic Web is a web for machines, but the process of creating and maintaining it is a social one. Although machines are helpful in manipulating symbols according to pre-defined rules, only the users of the Semantic Web have the necessary interpretive and associative capability for creating and maintaining ontologies. Ontology creation necessitates a social presence as it requires an actor to reliably predict how other members

[9]We performed a categorical core/periphery analysis with correlation optimization using UCINET 6 based on the connected part of the Flink social network data (N=528), available at `http://prauw.cs.vu.nl:8080/flink/graph`. The results show a clear C/P structure with 63 persons in the core and 465 persons on the periphery.

of the community would interpret the symbols of an ontology based on their limited description. With incorporating the notion of semantics into the web architecture, we have thus made the users of the system a critical part of the design.

We have argued elsewhere for a three layered view of the Semantic Web, namely the layer of communities and their relations, the layer of semantics (ontologies and their relations) and the layer of content items and their relations (the hypertext Web) [Mika, 2005b]. In this Chapter, we have formalized this view as a tripartite model of ontologies with three different classes of nodes (actors, concepts, and instances) and hyperedges representing the commitment of a user in terms of classifying an instance as belonging to a certain concept. We have shown the usefulness of this model by generating two kinds of association networks: the well-known co-occurrence network of ontology learning and a novel semantic network based on community relationships. Among the future work is the study of the two emerging social networks, based on object and concept overlaps.

The general advantage of the incorporation of the social context into the representation of ontologies is the possibility of studying emergence from user actions. Emergent semantics is likely to best complement well-established, but slowly evolving ontologies such as WordNet [Fellbaum, 1998], which lack the associative component.[10] We have also compared the two networks based on object and person overlap and noted the advantage of the second network: the possibility to extract semantics pertinent to a sub-community of the user network. In a sense, this is the opposite of mining general knowledge from search engines as in the work of Cimiano et al. or Etzioni et al. [Cimiano et al., 2004, Etzioni et al., 2004]. In comparison to these systems, our community-based ontology extraction has a great potential in extracting ontologies that more closely match the conceptualization of a particular community. For example, when trying to find associations between concepts used by the Web Services community, it is natural to consider only the associations created (explicitly or implicitly) by those involved in developing Web Services. As we have shown, by using this method the resulting ontology is more likely to be accepted as accurate by the community itself.

It seems that ontologies are us: inseparable from the context of the community in which they are created and used. A greater acknowledgement of this state —by incorporating the link between actors and concepts into the model of ontologies— have only benefits to bring in terms of more meaningful and easily maintainable conceptual structures. While we are only at the beginning of realizing these benefits, there is a clear magic as we see semantics emerge from the individual actions of a community at work.

[10] For example, according to WordNet the distance of the terms *Noah* and *ark* is quite large: their closest common ancestor in the hypernym tree is *object, physical object*. Yet, the Edinburgh master's students overwhelmingly associate the term *Noah* with *ark* and vice versa. The association is so strong in fact (78 and 79 percent of all terms mentioned in response, respectively) that it is safe to say that in the mind of the students these terms are solely defined by each other, in the context of the biblical story of Noah's ark.

Part IV

Conclusions

Chapter 11

The perfect storm

Hurricane Katrina formed as Tropical Depression Twelve over the southeastern Bahamas on August 23, 2005 as the result of an interaction of a tropical wave and the remains of Tropical Depression Ten. The system was upgraded to tropical storm status on the morning of August 24 and at this point, the storm was given the name Katrina. The tropical storm continued to move towards Florida, and became a hurricane only two hours before it made landfall between Hallandale Beach and Aventura, Florida on the morning of August 25. The storm weakened over land, but it regained hurricane status about one hour after entering the Gulf of Mexico. The storm rapidly intensified after entering the Gulf, partly because of the storm's movement over the warm waters of the Loop Current. On August 27, the storm reached Category 3 intensity on the Saffir-Simpson Hurricane Scale, becoming the third major hurricane of the season (see Figure 11.1). An eyeball replacement cycle disrupted the intensification, but caused the storm to nearly double in size. Katrina again rapidly intensified, attaining Category 5 status on the morning of August 28 and reached its peak strength at 1:00 p.m. CDT that day, with maximum sustained winds of 175 mph (280 km/h) and a minimum central pressure of 902 mbar. The pressure measurement made Katrina the fourth most intense Atlantic hurricane on record at the time, only to be surpassed by Hurricanes Rita and Wilma later in the season; it was also the strongest hurricane ever recorded in the Gulf of Mexico at the time as well (a record also later broken by Rita).

By August 26, the possibility of unprecedented cataclysm was already being considered. Many of the computer models had shifted the potential path of Katrina 150 miles westward from the Florida Panhandle, putting the city of New Orleans right in the center of their track probabilities; the chances of a direct hit were forecast at 17%, with strike probability rising to 29% by August 28. This scenario was considered a potential catastrophe because 80% of the city of New Orleans and 20% of the New Orleans metropolitan area is below sea level along Lake Pontchartrain.

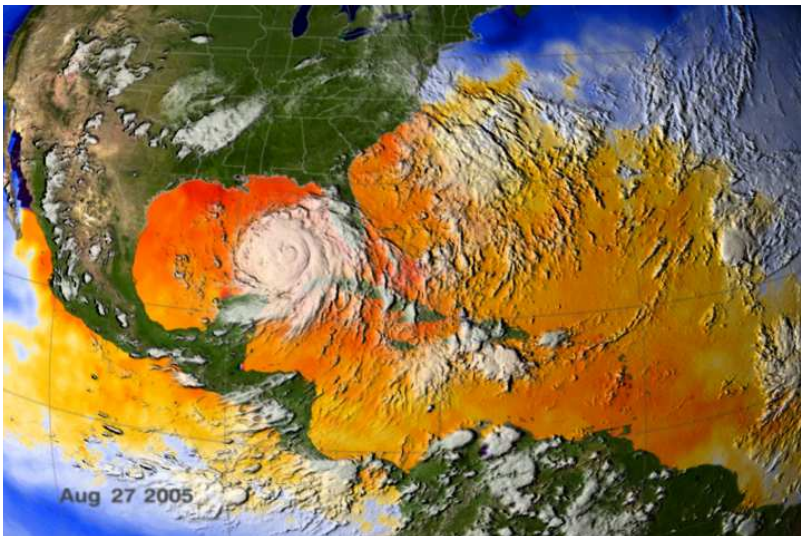


Figure 11.1: Satellite image of Hurricane Katrina on August 27, 2005. This image also depicts a 3-day average of actual sea surface temperatures for the Caribbean Sea and the Atlantic Ocean, from August 25-27, 2005. Courtesy of NASA.

> At a news conference at 10:00 a.m. on August 28, shortly after Katrina was upgraded to a Category 5 storm, New Orleans mayor Ray Nagin ordered the first ever mandatory evacuation of the city, calling Katrina "a storm that most of us have long feared". The city government also established several "refuges of last resort" for citizens who could not leave the city, including the massive Louisiana Superdome, which sheltered approximately 26,000 people and provided them with food and water for several days as the storm came ashore.
>
> *Source: Wikipedia, the online encyclopedia.*[1]

11.1 Looking back: the story of Katrina PeopleFinder

Hurricane Katrina was the deadliest US hurricane since 1928 and the costliest of all times. However, Hurricane Katrina was not only the perfect storm in a meteorological sense but in the way it exposed the failings of our infrastructure in times of emergencies. While the weaknesses of the physical infrastructure and the failings of the organizational structures of the emergency management are well documented, much less has been told about the story of the inadequacy of the current Web in dealing with emergency.

We are all familiar with the way levees and floodwalls designed to protect New Orleans have been breached, leaving 80% of New Orleans under water. Just as well known

[1] `http://en.wikipedia.org/w/index.php?title=Hurricane_Katrina&oldid=78783928`

are the failures of the rescue operations which have left people stranded on roofs for days. Those who reached the packed Superdome waited in almost complete darkness and without access to food, water, sanitation or fresh air. Those who survived described the Superdome as a concentration camp where rape, riots, armed violence and suicide were the order of the day.

Over time, however, the evacuees of the SuperDome joined the ranks of a total of 1.1 million who were displaced by Hurricane Katrina and moved to other shelters in the region. Shelters that varied in size from large to small, from the Houston Astrodome — where some 125 000 found temporary refuge— to housings set up in schools, churches and private homes.

However, it was unavoidable in the chaos of rescue and evacuation that families and friends were torn apart. As all of us would do the first thing the victims of Hurricane Katrina have done after finding shelter is to seek out their missing loved ones. As central coordination was lacking on the ground and communication options were limited, they invariably turned to the Internet for doing so.

Inevitably, evacuees have logged on to a number of different web sites for posting their notifications of missing persons and looking for survivors. The Mississippi coast area newspaper, the Gulf Coast News, for example started a Katrina survivor database as it was the natural source of news for the region.[2] Katrina survivor notices have also started to appear *en masse* on national bulletin boards such as Craigslist[3]. CNN started to collect submissions of notices from its viewers.[4] New discussion lists and forums for Katrina survivors have popped up by the minute (see Figure 11.2).

The problem with the resulting situation was blatantly apparent: how would family members, friend and co-workers find each other if they are all looking for information at different sites? The situation threatened to complement the physical suffering of the survivors with the emotional toll of being separated from their loved ones. Traditional search engines offered little help as they typically index information on a monthly basis, much too infrequent for websites that are updated by the minute. A central website put up by the authorities on time could have funnelled much of the information pouring in to a single database. But it could not have fully prevented the scenario that emerged. In fact, the diversity of sources is the result of those features of the Web that we cherish the most: a scalable, distributed design without a single point of failure that is open end-to-end and not controlled by any single authority.

In the absence of government action and just a few days after the disaster it was a volunteer effort that made the crucial steps toward a solution. A core group of bloggers and activists (most of whom have known each other from the Web and their volunteering in many previous efforts) started to band together using their blogs and IRC as a communication means.

---

[2]This database is still online at the time of writing and contains 77,000 records, see `http://www.gulfcoastnews.net/katrina/status.aspx`.

[3]`http://neworleans.craigslist.org/laf/`

[4]At one time, CNN's Katrina Safe List contained over a thousand pages of messages of safe and missing, although it has been taken off-line since then. See `http://www.cnn.com/SPECIALS/2005/hurricanes/list/`

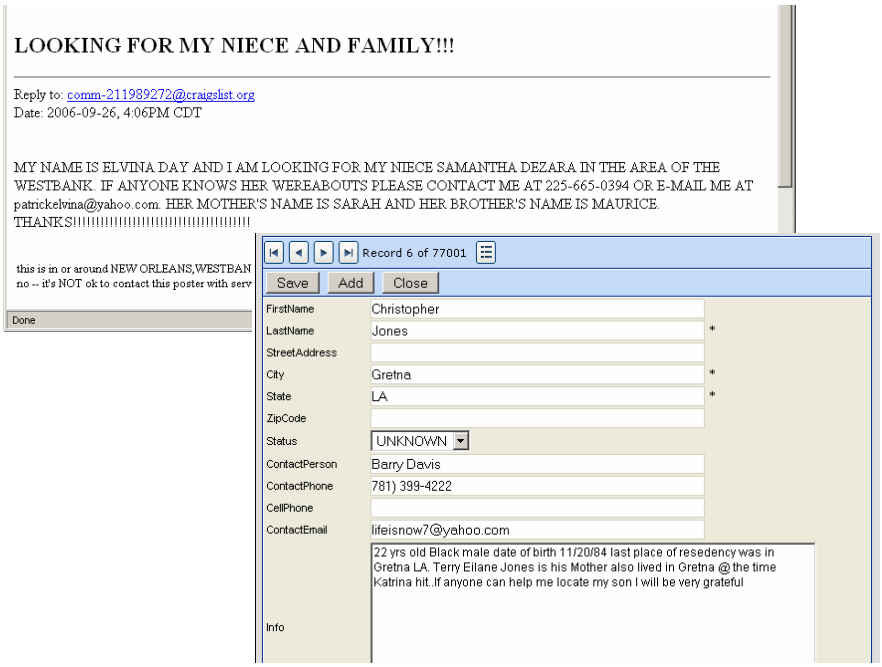


Figure 11.2: Survivor posting at Craigslist and the Gulf Coast News website.

On Friday, September 2 they came up with the key part of the technological solution: an XML format (called the PeopleFinder Information Format or PFIF) for records that describe missing or located individuals. The PFIF specification (written in less than 24 hours!) also defined the embedding of PFIF records in RSS feeds. The Salesforce.com Foundation committed to providing the back end database and search engine. The group also immediately begin to organize the social part of solution: they sent out urgent calls on volunteer mailing lists for people who would be willing to help in entering data manually into the database through a web form. They were also looking for people who would help to write scrapers to automatically extract content from other sites or to help other website owners write code to export data in PFIF format. They set up a Wiki[5] and mailing lists[6] to coordinate the work of data entry and technical development.

By Saturday, the first volunteers started entering data primarily from Craigslist. (Sites to be visited were collaborative tagged using the social bookmarking site del.icio.us.) As news of the project spread in the blogosphere it almost became a victim of its own success: the Wiki slows to a crawl as several hundred people are trying to edit the same page at times, the database needs to be taken down for update and the data collection

[5]The project Wiki is available at `http://katrinahelp.info/wiki/index.php/Katrina_PeopleFinder_Project`

[6]Archives of the Katrina developer mailing list are available at `http://orwant.com/katrina/by_date.html`

efforts bring other sites to a halt. There are also unexpected technical challenges along the way: for example, *chunking*[7] is problematic in cases like Craigslist where the content of the pages changes continuously and the posts are not numbered sequentially.

Nevertheless, problems are fixed immediately as volunteers work around the clock, even if the solutions are often practical hacks. The effort also learned to scale up organizationally and everyone in the community found a specific role, e.g. assisting newbies with questions, developing training material, gardening, maintaining the Wiki, chunking, promoting the project etc. There are 7,000 records by Sunday morning, and over 50,000 by the end of Monday. In time, over 4000 volunteers contributed and over 640,000 records are collected manually or automatically. A number of libraries and tools have been written in a variety of programming languages for dealing with PFIF data.

The Katrina PeopleFinder project has also spawned a sister project, the ShelterFinder. The ShelterFinder addresses the problem of collecting up-to-date information about shelters (their location, capacity, status etc.) Similar to personal status, this information is also scattered around the Web in case of emergencies or passed along between agencies in various formats such as Excel sheets. ShelterFinder brings this data into a central search facility and also integrates geographic visualization (GoogleMaps) to make it easier to find shelters by location.

While ShelterFinder still exists, the Katrina PeopleFinder project came to an end before the end of the month just as quickly as it started. The data that has been collected has been merged into even larger databases set up by Google and a co-operation between the American Red Cross and Microsoft and eventually almost all victims were reconnected.[8] Much of survivor data on the Web have been also removed[9] due to privacy concerns once the emergency was gone. The project itself was mired in a legal debate already after two weeks of existence. One of the sources scraped, the Gulf Coast News (GCN) has sent a cease and desist letter to the project citing copyright violation. Katrina turned out to be also a business opportunity for many who have seen the traffic of their websites (and their advertising revenue) increase in the wake of the disaster.

The emergency wasn't gone, however. Hurricane Rita, the fourth most intense Atlantic hurricane ever recorded hit Florida less than a month after Hurricane Katrina. Hurricane Wilma, the most intense Atlantic hurricane ever recorded followed Rita on October 18, 2005. With Hurricanes Wilma, Emily, Katrina, and Rita, 2005 became the first year on record in which four Category 5 hurricanes developed in the Atlantic basin.

11.1.1 The Semantic Web

The Katrina PeopleFinder is a remarkable project because it has shown a glimpse of the power in combining novel web-based technologies and civil activism.[10] Some call

[7] The technical term invented by the community for the breaking up of large sites into portions manageable by a single volunteer.

[8] None of these databases are available any more, although by far not all have returned to New Orleans and certainly not all have found each other: even after a year there are still friends and former co-workers looking for each other on Craigslist and other remaining sites.

[9] Removed from its original location that is...

[10] Unfortunately, the story and the lessons from the Katrina PeopleFinder project have not been widely publicized. The best records are the blog entries from some of the core members, including Ethan Zuck-

this idea *Recovery 2.0* (a combination of Web 2.0 and recovery efforts), others term it *social source*, the mix of social support and open source. (As one of the founding project members have phrased it: "Sometimes code is the solution. Sometimes 2,000 loosely organized people are the solution."[11]) As a result of the efforts of those involved an immeasurable number of people were saved from the emotional toll of being separated from their loved ones.

Yet those involved on the technology side of the project are the first to admit that they tackled an important problem with inadequate tools. However, an emergency is not the time to innovate or educate: as one of member phrased it "the technology has to be pre-positioned, accessible, and you can't need to 'ask permission' or even involve the folks that 'own'/maintain the technology to use it for your purposes." Needless to say, most of the developers involved in the project were only marginally aware of semantic technologies if at all, which has yet to find its way to mainstream web developers.

The reader who has read this book, however, knows the difference that semantic technologies could have made. In the following we highlight two key issues where the technologies discussed in this book might have made a significant impact on the capabilities of the system.

Reducing interdependence on the schema level

The choice for XML in the project was a conscious technical decision due to the wide availability of XML tools, e.g. XML Schema validators that could be used to validate data against the XML Schema contained in the PFIF specification. XML as an interchange language was ideal in connecting disparate systems on a syntactic level.

The reliance of XML schema languages, however, introduced a significant dependency in the design. In particular, after the initial release of PFIF standard project members have quickly found out that their XML schema is far from perfect: some sites collect much more detailed information than the simple contact fields of the PFIF schema. But there was no way to change the schema any more: by that time a number of other sites and a host of tools were supporting it. The dependency of these tools on a particular version of the PFIF format created a dependency that had to be continuously reckoned with from then on.

And in an emergency all dependencies are costly: ideally actions need to be parallelized as much as possible. Dependencies between developments introduce delays through the necessary communication involved. In the case of PFIF, this would have essentially entailed notifying all developers and synchronizing the switch to the new schema. The costs in terms of time loss would have been so high that introducing a second version of the schema was never even considered. Instead additional information was shoe-horned into plain text fields where their semantics was completely lost. In essence, this information could not be searched other than by keywords.

We know that RDF and OWL would have offered a way to remove much of this interdependency through greater flexibility in modelling. RDF offers all the advantages

erman `http://www.ethanzuckerman.com/blog/?p=170` and David Geilhufe `http://socialsource.blogspot.com/2005/10/personal-history-of-katrina.html`

[11] `http://www.ethanzuckerman.com/blog/?p=170`

of XML in terms of shared syntax and tool support is improving rapidly. More importantly, web-based knowledge representation languages are prepared for exactly the kind of scenario that is present on the Web, in particular supporting the independent, parallel evolution of schemas (ontologies). For example, adding a new field such as birthdate to the definition of the concept *foaf*:*Person* can be done independently of the maintainers of FOAF schema. Further, it is possible to describe this new notion in such a way that other parties will have at least a partial understanding of it. For example, we could state that the value of the birthdate property is a point in time (a date). Other agents would then at least know that a birthdate can be visualized along a timeline, it can be compared with other kinds of dates etc.

Other implementors may take over our use of the term birthdate, but they may also come up with their own separate property for exporting birthdates. In particular in cases where an existing database is converted to PFIF or some other interchange format, it saves time and effort to export the database using the original database schema.[12] If that happens, two properties with the same semantics can still be mapped to each other either manually or automatically (*ontology mapping*), allowing processing agents to understand that two or more properties have the same intended meaning and treat them uniformly.

An ontology-based approach would have also made it easier to combine the collected data with available sources of background knowledge and web services. For example, in order to visualize shelters on a map in ShelterFinder there was a need to locate the geographic coordinates of the shelters. Such a task could have been done by combining shelter data with the freely available database that contains the mappings from US zip codes to geo-coordinates. Standards such as the SPARQL protocol and query language allow to query such data sources remotely, without the need of adding them to the local knowledge base. For more precise location, the system could also have been more easily connected to external geo-locator services that return geographic coordinates based on place names or complete addresses.

The kind of bottom-up, emerging ontologies (*folksonomies*) that we have seen in Chapter 10 could have also played a key role in breaking down the complexity of the emergency management task. Folksonomies represent a different trade-off in formality versus control than manually engineered ontologies. Tagging is a simple activity that even those unaware of ontologies can easily do as shown by the success of many folksonomy systems in Web 2.0 applications. Tagging systems can thus be built in the user interface of web applications without adding much complexity.

Tagging also offers much more flexibility than traditional ontologies at the cost of loosing explicit semantics. In the PeopleFinder project, for example, as the project went along the need came up to mark records that need correction or due for removal. As this need was not anticipated there wasn't an appropriate field in the schema of the project. Using tagging, however, such records could be easily tagged with some specific keyword using the interface, without even the need to adjust the underlying ontology. Once the action has been completed, the tag could have been removed just as easily. In this case, the semantics of the tag only needed to be understood by the project members and only

[12] In fact, tools such as PHPMySQLAdmin allow the export of a database directly to XML. With similar tools for RDF, there would be no coding effort involved in making data available in RDF.

by those concerned with gardening and thus the loss of explicit semantics would have been an acceptable price to pay for increased flexibility and user convenience.

In summary, emergencies have the inherent property that they can cannot be completely planned for. In such cases technologies that offer the greatest flexibility will allow to adapt to changing conditions with the least delay. The RDF and OWL languages introduced in this book have been specifically designed for knowledge representation in web-based settings, which could have been exploited in the Katrina PeopleFinder to reduce the interdependency between data and services maintained by project members or third parties.

Aggregation of social individuals

PeopleFinder served as a search interface to a collection of data gathered from the Web. The system was designed in a way that users (evacuees themselves) would be expected to come to the site searching for their loved ones among the announcements of missing and safe persons.

However, the system was not able to aggregate information or automatically match descriptions of missing persons against descriptions of safe persons. Using the technologies we have described in this book it would have been very well possible to automate the tasks of merging identical records and matching descriptions of persons. Ultimately, the system could have brought families and friends directly in contact after matching their records instead of waiting for them to search for each other.

We have seen in this thesis what is needed for that in terms of technology. RDF as a knowledge representation language offers a good starting point as it provides a globally unique mechanism to identify resources. We have shown how to use OWL and rules to describe in a specific domain what it means for two things to be equal and how reasoners can be applied to the task of instance unification or *smushing* (see Chapter 6). We have implemented the general framework for this method in the Elmo API and used it to disambiguate person references in the Flink system which provided the data for our research on the Semantic Web community (see Chapter 7).

When I joined the ShelterFinder project I quickly realized that it was not the time to educate the volunteer community about the Semantic Web. However, the costs of not using any automated technology for removing duplicate records was also painfully clear. In this project, duplicate records of shelters were either found by searching the database (and noticing the duplicates) or by sorting Excel sheets on certain columns. It was clear that while the first method relies on chance and human effort, the second is hardly foolproof as it only works if prefixes of strings overlap. This was a clear case where a little semantics went a long way. I have donated a converter from CSV format (supported by Excel) to RDF and wrote a simple smusher that matched the names of shelters based on string similarity. While hardly rocket science in terms of research, it could be used by the project in filtering out already known shelters from incoming Excel sheets.

The reader may note that before semantic technologies could have been brought to bear in matching descriptions of missing persons with reports of safe persons, the PeopleFinder project would have also needed to enrich their schema with personal characteristics that could have been used for matching. (As discussed in the previous section

the PFIF was rather limited and could not be extended due to necessary freezing of the schema.) An additional problem the project would have needed to deal with is that many of the records imported from other sites contained person descriptions in natural language (see for example the postings in Figure 11.2). To some extent automatic information extraction techniques could have been applied to extract some of this metadata. As such techniques are hardly fool-proof, it is likely that human effort would have been necessary to check and eventually re-enter some of the metadata. However, what the project has also shown is that such help is nearby when it's needed the most.

11.1.2 Social Networks

Much of the help the PeopleFinder project received came through the online social networks of activists. Thanks to internet-based communication technologies such networks were promptly activated as the situation escalated and served as a critical factor in mobilizing resources for the project. Evacuees themselves also tried to reconnect their personal networks. In fact, the records of PeopleFinder are not only descriptions of individuals but rather descriptions of two persons —the one creating the note and the one addressed— and their relationship.

Conceptualizing the system as a social network application would have allowed to make connections between people who could not be directly connected to each other, but could be reached through a third person. Studying the data using the methods of Social Network Analysis would have made it possible to gain important insights into how precisely networks are used in times of emergencies, allowing us to build even better applications for crisis management.

Although not a discussion that have entered in detail, we would also note the urgent need to clarify the legal and ethical boundaries of carrying out social science research on the Web. The story of the Katrina PeopleFinder serves as a reminder of the possible collision between the public interest on side and the law and the markets as regulators on the other side.

When aggregating data from multiple electronic sources one is particularly liable to accusations that sensitive personal data has been used in ways that were not intended (and could not have been foreseen) when making some information available on the Web. Collecting information from the Web is also likely to cross the boundaries of national legislations and ethical guidelines. With respect to our web mining method we can also state that asking permission for the reuse of every individual piece of information on the Web would not have been a real possibility either. Such possibilities have not even been foreseen in existing ethical codes for Social Science research that involves the handling of personal data.

Networks in science

Our research has focused on networks that form under less strenuous circumstances: we have studied the networks of researchers working on realizing the Semantic Web. Our inspirations, however, have been similar: understanding the role of networks in science is an important first step toward organizing the scientific process in more efficient ways.

The scientific domain offers an ideal terrain to demonstrate our methodology for social network analysis based on electronic sources. Not only researchers are present on the Web and carry out discussions on electronic forums of all kinds but the very objects produced by science itself are undergoing a thorough digitalization. This means that more and more of science is visible through the Web and can be manipulated by computers.

The move towards *e-science* is clearly visible, for example, in the changing role of libraries and publishers, the traditional facilitators of the scientific process. The library system is rapidly turning into networks of repositories of digital objects instead of books and periodicals, while publishers more and more require scientists to publish their data in electronic formats alongside with the text of their articles. Ultimately, the scientific article itself will be decomposed into its two main constituents (data and argumentation) so that we can analyze the data separately and apply various argumentations to it just as we can apply different stylesheets to webpages today.

That social networks matter in science is also an important lesson that everyone who has done a PhD would learn. In the case of PhD students, collaborating with more senior researchers is the primary way a PhD student would acquire the necessary skills to write successful articles. And because reputation is an important organizing factor in communities with no formal rules, it is an advantage to co-author with more senior researchers. It is equally important to build (preferably) direct relationships with those key members of the research community who have roles in managing journals and conferences, the outlets of scientific production. (This has informational advantages but may also result in some minor favors such as the softening of a deadline.) Lastly, for more senior researchers social networks are also intensively activated when acquiring research funding: some funding agencies such as the EU allow only multinational research teams to apply and both the assignment of total funding to certain research areas and the individual grant decisions are determined by experts who are necessarily members of the community.

In our case we have been interested in the impact of social networks on scientific outcomes as measured by publication performance. In particular, we would test the common intuition that the most successful works come about by combining one's own ideas with that of researchers at different research groups, possibly in different specialties. We have proved that indeed a cognitive diversity in the personal networks of researchers has benefits beyond the well-known structural advantages of large and sparsely connected networks (see Chapter 9). While this has been a traditional study in terms of analysis, we introduced a set of novel methods in the representation and management of social network data. In particular, our semantic-based representation of network data made it possible to automatically aggregate a number of electronic sources that contained information about the relationships in our target community, the community of researchers working on the Semantic Web. Among others, we have exploited the content of Web pages for social network mining, which provided us with plentiful data for analysis. Our automated methods allowed us to scale up our effort significantly when having to deal with multiple data sources containing information about the same set of social individuals and their relationships. Further, the possibility to use multiple data sources allows to design studies that are more robust and allow multiple viewpoints on the social structures within a community.

Beyond our own study, semantic technologies offer advantages to the whole of *e-social science*. Shared ontologies for representing data in a scientific domain allow researchers to exchange and reuse data more effectively. Just as it is already common in some of the natural sciences, there are also attempts in the social sciences to set up repositories of research data instead of repositories of publications. Such repositories will need representations that are formal (machine processable), commonly agreed upon yet flexible enough to serve a variety of viewpoints on what is the specific object of a study and how it is to be performed.

As a benefit on the side, the necessary formalization of social science concepts will no doubt lead to important discussions. Although Social Network Analysis is one of the most formalized field of Social Science, there are still significant differences in the definition of some of the very basic concepts of the field as we have seen from the many ways to capture the idea of tie strength. Formalization can lead to a clarification of the existence of these different viewpoints, what measures they induce and how these measures ultimately relate to each other.[13]

11.2 Looking ahead: a Second Life

There is only one area that captures most succinctly the mesmerizing opportunities and the mind-boggling challenges ahead of Social Networks and the Semantic Web. This is the area of artificial worlds.

Systems such as Second Life present alternate realities buzzing with life.[14] Second Life is a simulated environment populated by human controlled avatars; three-dimensional characters in a three-dimensional environment. The virtual reality of Second Life is inhabited by over 800,000 avatars (the size of Amsterdam) at the time of writing[15], who live their lives in an environment where flying is normal and nobody dies. Second Life caters to a primitive desire we most likely carry since we gained consciousness: what if we could step out of our self and redefine who we are?

Second Life allows just that by allowing to customize our self and in fact, allowing us to have as many personae (represented by different avatars) as we like. Yet Second Life is *realistic* in many respects and certainly more serious than to be described as a game. For example, Second Life has a striving economy based on the Linden Dollar, which is freely convertible to US dollars. Just in the last 24 hours 350,000 US dollars have been spent in Second Life. In fact, many residents of Second Life have already given up their real world jobs to make a living in the alternate reality by selling, for example, virtual clothing, teaching classes, by acting in movies shot in the virtual reality or by performing music in bars. (Needless to say, one has to be really good to get into the hottest venues.) The company that runs Second Life earns money by selling real estate in the virtual world; but they are not the only one: BusinessWeek devoted a recent cover

[13] For a broader view on the mutual influence of Computer Science and Social Science methodologies see also [Akkermans and Gordijn, 2006a, Akkermans and Gordijn, 2006b].

[14] `http://secondlife.com`, `http://lindenlab.com`

[15] October 1, 2006

story to Anshe Chung, who earns hundreds of thousands of (actual) dollars as the most prominent real-estate mogul in Second Life [Hof, 2006].

Most of all, however, Second Life residents socialize, build relationships and yes, have sexual affairs in specifically marked Adult areas. (Although Second Life policy prohibits giving out real identities, there have been real world marriages that started with a relationship in SL.) Not only can they talk to each other, but also use gestures and body languages (see Figure 11.3). And in this new level of interaction lies a grand new Challenge for AI: how to introduce machine intelligence into the world of Second Life?

Figure 11.3: Who said they can't dance? While some sit around the bar, some Second Lifers take on the dance floor at a party thrown by the Second Life Herald, SL's in-game newspaper covering the politics, society and events that occur inside Second Life.

Surely, a machine that would pass the Turing test would not be considered realistic enough for a Second Lifer: a computer standing behind a curtain would be promptly exposed as a cheat no matter how wittily it answers our questions. In order for our computer driven avatar to be accepted it would certainly need to have a social life.

Social Network Analysis, the study of the structure, formation and impact of social networks could come to the rescue. But Second Life could teach just as much to Social Network Analysis. Doing social science research through a researcher-avatar in the virtual reality of Second Life would be certainly low cost (no need to leave the room) and fast (hey, you can fly from subject to subject!). Such research could no doubt also rely on electronic data (e.g. collected by intelligent objects placed in the environment) and would need to be liable to the same questions as Social Science research on the Web, i.e. how much of what we learn in the VR world holds for the real world. If that is what we

are trying to understand... it is also very well possible that SNA would find applications inside Second Life, e.g. for predicting the effects of word of mouth marketing campaigns under way by some major corporations (real and un-real).

Systems like Second Life may also mark the future of the Web as they can be easily integrated with Web content. Many applications may directly benefit from being conceptualized in the virtual space. Not only real world retailers such as Amazon would be interested to show their merchandize, but also projects like the Katrina PeopleFinder could be built in a special part of Second Life. In this space Katrina evacuees would get together and meet rescue workers, explore shelters laid out on a simulated terrain of the real disaster area etc. Creating the application in a 3D reality would play to our inherent strength of processing information faster when presented in a familiar context of space, using human interactions instead of written text.

There are also ample opportunities for improving our understanding of semantics. Part of the difficulty in developing technologies for the Semantic Web is that ultimately the truth of statements (and thus the correctness of inferences) is determined against an external reality: *the truth is out there*. As a consequence, for example, evaluating ontology construction or ontology mapping methods is a human-complex problem. We as humans can determine (or agree upon after some discussion) on some golden standard based on what we believe is true about the segment of the world to be modelled. We can then evaluate our algorithms against this understanding and design new ones that match our intuitions better.

The interesting aspect of Second Life is that *the truth is inside the system* at least to some extent. The statement "people can't fly" is true in our reality but easily falsified in Second Life *even by a machine*. (If only we equate avatars with people and apply our usual definition of what flying is.) Machines in Second Life has a much greater access to the grounding of statements, which in theory enables them to be much more intelligent in their own reality. Second Life will probably also vividly demonstrate our statements about the socially constructed nature of ontologies. Second Life residents will no doubt create many abstract, socially-constructed concepts expressed in words whose meaning is relative to some or all Second Life residents.

In fact, that "people can fly" is not in conflict with our own mental model as we know that such semantics is relative to the context of Second Life. There is some danger that we loose this grip if we immerse too much in the virtual reality: cognitive science tells us that when faced with an observation that contradicts our own beliefs we have two basic choices: reject the observations on some grounds or adapt our mental model. The theory of cognitive dissonance dictates that we will choose the option that can most easily reduce the tension [Festinger, 1957]. If the virtual reality is compelling enough (or we spend long enough time there) would there be a danger that we adopt our mental models and believe that people can fly?

That is a scary thought. But here is happy one: I turned 19 yesterday. That is: one of my avatars has turned 19. It was a hell of a party... with lots of friends and *all of me*.

Samenvatting

Zelfs terugkijkend is het moeilijk te zeggen of wij het Web hebben veranderd of dat het Web ons heeft veranderd.

Maar hoewel het proces een vraag is, de veranderingen zijn een feit. Een recent grootschalig onderzoek naar het Internetgebruik door Amerikanen heeft de enorme verandering laten zien in de manier waarop men de online wereld benadert [Boase et al., 2006]. Wanneer we het Web beschouwen als een enorm mededelingenbord, dan plaatste in het begin soms iemand eens een krabbel, maar men liep er vooral langs heen, turend van de ene naar de andere mededeling. Tegenwoordig 'surfen' we niet alleen meer, maar is het Web een verlengstuk van onszelf geworden dat ons helpt om anderen te bereiken. We hebben geleerd om het reclamebord te gebruiken om actief anderen te zoeken en hebben het zo een ontmoetingsplaats gemaakt. Rondom het bord discussiëren we, vragen en geven we advies en delen we onze smaken, ervaringen en ideeën met vrienden en onbekenden, en doen op deze manier nieuwe contacten op.

Het resultaat blijkt duidelijk uit het onderzoek: de contacten die we opdoen gebruiken we zelfs in situaties waarin we vroeger alleen een beroep deden op nauwe verwanten, goede vrienden en familie. Het Internet heeft de grootte en samenstelling van onze sociale verbanden veranderd: onze verbanden zijn uitgebreid met een heel veel 'vage kennissen', d.w.z. al die bekende gezichten in onze *blog rolls*, *buddy lists*, *chat groups*, *web fora*, mailinglijsten en de vele andere fora waarin we met anderen communiceren.

Het moge duidelijk zijn dat het mededelingenbord mee moest veranderen om zich aan zijn nieuwe functie aan te passen. De term *Web 2.0* wordt tegenwoordig gebruikt als overkoepelende term voor al deze geleidelijke veranderingen. Ten eerste is het Web eindelijk het lees èn schrijf Web geworden zoals zijn uitvinder het oorspronkelijk bedoeld had: de nu populaire plaatsen zijn door en voor mensen gemaakt. De kennis van de massa wordt gebruikt om grote kennisbronnen te bouwen en onderhouden, zoals de online encyclopedie Wikipedia. Maar die 'massa' schrijft niet alleen mee aan een encyclopedie, zij deelt ook foto's en muziek, jaagt op nieuws en boeken, ontwerpt software, schrijft verhalen, en nog veel meer. Het Web heeft zich technisch gezien ook moeten aanpassen. Door gebruik te maken van AJAX technologie voelt de interactie met websites veel natuurlijker aan, terwijl *RSS feeds* en andere technieken zorgen voor een betere verbinding tussen de inhoud van het Web en de gebruikers. De enorme populariteit van *scripting* programmeertalen suggereert dat ook het programmeren is gedemocratiseerd en nu geworden is tot een kunst in plaats van een ambacht. Tenslotte, het gevoel van (gemeenschappelijk) bezit blijft mensen stimuleren om creatief te experimenteren met

de inhoud van het web in de vorm van *mash-ups*, web toepassingen die door gebruikers gecreëerde informatie uit verschillende bronnen combineert.

In tegenstelling tot het Web 2.0 is het Semantisch Web een meer bewuste inspanning onder leiding van het *World Wide Web Consortium* (de standaardisatie organisatie van het Web) om het Web toegankelijker te maken voor machines. Op dit moment is de meeste informatie op het Web alleen toegankelijk voor mensen. Het Semantische Web voegt een aantal extra lagen toe aan de architectuur van het Web die het mogelijk maken de inhoud van het Web te beschrijven met behulp van gemeenschappelijk vocabulaires, ook wel ontologieën genoemd. Dit zou computers in staat moeten stellen te redeneren over de kennis die in de ontologieën is uitgedrukt, bijvoorbeeld door relevante informatie uit verschillende bronnen te combineren en daaruit conclusies te trekken op een manier die lijkt op hoe mensen zouden redeneren. Hoewel het Semantisch Web een infrastructuur voor machines is, zal de kennis waaruit het is opgebouwd en de afleidingsregels waarmee geredeneerd kan worden uiteindelijk door mensen moeten worden geleverd. Kortgezegd is er geen semantiek zonder mensen, waardoor het Semantische Web net zo goed een sociaal als een technologisch systeem is.

Deze ontwikkelingen zijn zowel interessant voor onderzoekers in de sociale wetenschappen en de informatiekunde, als voor ontwikkelaars van sociaal semantische software voor het Web. Aan de ene kant biedt het zich ontwikkelende *Sociale Web* onverwachte mogelijkheden om sociaal gedrag te observeren door de interactie op het Web te volgen. Aan de andere kant vraagt de metadata die door gebruikers is aangeleverd om een andere behandeling dan de andere informatie. De door gebruikers aangeleverd informatie kan namelijk niet los gezien worden van de sociale context waarin het is gegenereerd. Met deze kennis, met name de sociale netwerken van gebruikers, kunnen de machines ook weer redeneren. Dit biedt ongezien mogelijkheden om informatie systemen te bouwen die rekening houden met de sociale context. Voor het Semantische Web betekent dit in het bijzonder dat intelligente toepassingen kunnen worden gebouwd die rekening houden met het sociale karakter van betekenis.

In dit boek beschrijven we twee grote *case studies* waarin beide mogelijkheden gedemonstreerd worden. De eerste studie laat zien hoe een onderzoeksgemeenschap via de interactie op het Web in kaart kan worden gebracht en geanalyseerd, waarbij ook gebruik wordt gemaakt van andere databronnen, zoals e-mails en publicaties (hoofdstuk 9). Het distilleren van een sociaal netwerk uit de informatie op het Web speelt in deze studie een grote rol. De grootschalige en dynamische gegevens over het netwerk die deze methode levert zouden niet met enquête methoden verzameld kunnen worden. Andersom is de semantische technologie essentieel voor het representeren en aggregeren van informatie vanuit verschillende bronnen. Methoden uit de sociale netwerkanalyse worden gebruikt om voorspellingen over de prestatie van onderzoekers te genereren.

Omdat de methoden die we gebruiken breder toepasbaar zijn dan in ons onderzoek over een wetenschapsgebied besteden we meer aandacht aan het beschrijven van de methoden dan aan het bespreken van de resultaten. In hoofdstuk 4 vatten we de mogelijkheden van het (her)gebruik van elektronische gegevens voor netwerkanalyse samen. Hoofdstuk 8 bevat een gerelateerd onderzoek waarin twee methoden voor het verkrijgen van sociale netwerk-gegevens vanuit het Web worden geëvalueerd. De semantische technologie voor het aggregeren van sociale netwerkgegevens wordt beschreven in de hoofdstukken 5

en 6. Tenslotte beschrijven we in hoofdstuk 7 hoe onze methoden zijn geïmplementeerd in het onderscheiden Flink systeem. Deze beschrijvingen maken het niet alleen mogelijk om onze resultaten te reproduceren, maar ook om onze methoden in verschillende soorten situaties toe te passen. Hiervoor zullen de methoden aan de andere situatie en informatiebronnen moeten worden aangepast, maar blijven de voordelen van het volledig geautomatiseerde proces gebaseerd op elektronische data behouden.

Ons tweede onderzoek belicht de rol van de sociale context in door gebruikers gecreëerde classificaties van inhoud, in het bijzonder in zogenaamde *tagging*-systemen, ook wel *folksonomies* genoemd (hoofdstuk 10). *Tagging* wordt vaak gebruikt om de inhoud van veel Web 2.0 diensten te ordenen, bijvoorbeeld in del.icio.us, het systeem voor gemeenschappelijke bladwijzers op het internet, en Flickr, de website die met mogelijk maakt om foto's te delen. Wij beschouwen *folksonomies* als lichtgewicht semantische structuren, waarbij de semantiek van de *tags* (labels) na verloop van tijd boven komt drijven door de manier waarop de labels worden gebruikt. Voor het bestuderen van de *tagging*-systemen gebruiken we de concepten en methoden van de netwerkanalyse. We laten zien dat *folksonomies* inderdaad een veel rijkere semantiek hebben dan op het eerste gezicht lijkt en dat deze afhankelijk is van de sociale context van de applicatie. Deze resultaten zijn in het bijzonder nuttig voor het ontwikkelen van het Semantische Web via een *bottom-up* aanpak, waarin de samenwerking tussen mensen een belangrijke rol speelt. Het plaatsen van de beschikbare kennis in een sociale context maakt ook de weg vrij naar gepersonaliseerde toepassingen, zoals sociaal zoeken.

Zoals duidelijk wordt uit bovenstaande beschrijving worden beide studies gekenmerkt door de interdisciplinaire aanpak, waarbij concepten en methoden vanuit de Kunstmatige Intelligentie worden gecombineerd met die vanuit de Sociale Netwerk Analyse. Het is echter niet noodzakelijk om voorkennis over deze gebieden te hebben, de hoofdstukken 2 en 3 bevatten de benodigde inleidingen. Hierdoor zou ons werk zowel toegankelijk moeten zijn voor sociale wetenschappers met interesse in elektronische data als informatiekundigen met interesse in sociaal-semantische toepassingen.

Ons belangrijkste doel is niet om beide gebieden in detail te behandelen, maar om zowel sociale wetenschappers als informatiekundigen inzicht te geven in de concepten en methoden die buiten hun eigen gebied vallen. We laten iets zien van de voordelen die dit begrip kunnen bieden voor het oplossen van de complexe problemen die een inherent interdisciplinair karakter hebben. Onze hoop is dat we mensen inspireren om verdere creatieve experimenten uit te voeren die ons een beter begrip zullen geven over zowel de sociale interactie die online plaatsvindt als over de eigenschappen van menselijke kennis. Dit begrip is onmisbaar in een wereld waarin de afstand tussen beide disciplines naar verwachting steeds kleiner zal worden door de online omgevingen die een steeds grotere rol in ons sociale leven gaan spelen, zoals de virtuele werelden van *Second Life*. Alleen met dit goede begrip zullen we in staat zijn om systemen te ontwerpen die zowel in hun redenen als in hun sociale mogelijkheden echte intelligentie laten zien en ons zo kunnen leiden door een steeds complexere online wereld.

De auteur bedankt de Vrije Universiteit Research School for Business Information Sciences (VUBIS) voor de steun bij het uitvoeren van het onderzoek in dit boek.

Summary

Whether we changed the Web or the Web has changed us is difficult to distil even when equipped with the wisdom of hindsight.

While the process is a mystery, the changes are a fact. A recent large scale study on the Internet use of Americans has recorded the dramatic shift in the way that we approach the online world [Boase et al., 2006]. If we think of the Web as a giant billboard, we can say that the early days were spent with some affixing notes to this board, while most were merely passing by, carelessly surfing from one note to the next. These days however we do not just 'surf' anymore. We have learned to use the billboard to actively seek out others and made it a gathering place. Around the board we discuss matters with friends and unknowns, ask and give advice, share our tastes, experiences and ideas, and build relationships in the process.

The end result is clear from the survey: the ties we build are activated even in those situations that in the past we used to solve solely with the help of our closest allies, our intimate friends and family. The Net has changed the size and composition of our social networks: in particular, our networks have grown with an array of weak ties – a common name for those familiar faces on our blog rolls, buddy lists, chat groups, fora, mailing lists and the myriad other forums of our interactions.

Needless to say, the billboard had to change to adapt to its new function. What we now call Web 2.0 is a collective name for these evolutionary changes. The most important of these changes is conceptual: the Web has finally become the read/write Web that its inventor originally intended it to be. The popular 'places' of today are created by and for the people. We not only hang around the billboard and socialize, but use the space of the board to build a collective intelligence. The wisdom of the crowd is put to use in building and managing large repositories of knowledge such as Wikipedia, the online encyclopedia. And 'the crowd' is not only editing encyclopedias either, but we are also sharing photos and music, hunting for books, filtering news, writing stories, organizing digital collections through classification, and much more.

Technological change played the minor role in this process and it mostly had to with the adapting the Web to collaboration. New technologies such as AJAX-based development and a no-nonsense design of the websites have significantly improved the user experience while interacting with web applications, while RSS feeds and other technologies improved the connectivity between users and the content of the Web. The immense popularity of scripting languages hints at the way programming has been democratized and turned into a form of art, rather than engineering. The (sense of) collective own-

ership of both content and code continues to inspire creative experimentation with web content in the form of mash-ups, web applications that combine user generated content from multiple services.

In contrast to Web 2.0, the Semantic Web is an effort to carry out a more fundamental change in the architecture of the Web. Initiated by Tim Berners-Lee, the head of the World Wide Web Consortium (the standards organization behind the Web), the idea of the Semantic Web is to make the Web friendlier for machines. While at the moment most of the content in the online world is only accessible to human readers, the Semantic Web would provide additional layers of Web architecture for describing content using shared vocabularies called ontologies. This would allow computers to reason with the knowledge expressed in Web resources, in particular to aggregate relevant information from multiple sources and to come to conclusions in a manner that resembles human logic. While an infrastructure for machines, the knowledge that fills the Semantic Web and the rules of reasoning will in fact be provided by humans. In short, there is no semantics without humans and this makes the Semantic Web as much a social system as a technological one.

These developments are of interest to researchers in both the Social and Information Sciences, as well as to practitioners developing social-semantic software for the Web. On the one hand, the emergence of the Social Web opens up never foreseen opportunities for observing social behavior by tracing social interaction on the Web. Semantic Web technology comes to help by providing the means to aggregate the fragmented information about our online social networks. On the other hand, user generated content and metadata in social software requires a different treatment than other content and metadata. In particular, this knowledge comes with additional information about the social context in which it is conceived and this information (in particular, the social networks of users) is also accessible for our machines to reason with. This provides unprecedented possibilities in building socially-aware information systems.

In this book we provide two major case studies to demonstrate each of these opportunities. The first case study shows the possibilities of tracking a research community over the Web, combining the information obtained from the Web with other data sources (publications, emails) and analyzing the results (Chapter 9). Social network mining from the Web plays an important role in this case study for obtaining large scale, dynamic network data beyond the possibilities of survey methods. In turn semantic technology is the key to the representation and aggregation of information from multiple heterogeneous information sources. The methods of social network analysis are applied to the results in order to obtain network-based predictors of the performance of researchers.

Our methodology is more generally applicable than the context of our scientometric study, and thus we will spend significantly more time on describing our methods than discussing our results. We summarize the possibilities for (re)using electronic data for network analysis in Chapter 4 and evaluate two methods of social network mining from the Web in a separate study described in Chapter 8. We discuss semantic technology for social network data aggregation in Chapters 5 and 6. Lastly, we describe the implementation of our methods in the award-winning Flink system in Chapter 7. In fact these descriptions should not only allow the reader to reproduce our work, but to apply our methods in a wide range of settings. This includes adapting our methods to other

social settings and other kinds of information sources, while preserving the advantages of a fully automated process based on electronic data.

Our second study highlights the role of the social context in user-generated classifications of content, in particular in tagging systems known as folksonomies (Chapter 10). Tagging is widely applied in organizing the content in many Web 2.0 services, including the social bookmarking application del.icio.us and the photo sharing site Flickr. We consider folksonomies as lightweight semantic structures where the semantics of tags emerges over time from the way tags are applied. We study tagging systems using the concepts and methodology of network analysis. We establish that folksonomies are indeed much richer in semantics than it might seem at first and we show the dependence of semantics on the social context of application. These results are particularly relevant for the development of the Semantic Web using bottom-up, collaborative approaches. Putting the available knowledge in a social context also opens the way to more personalized applications such as social search.

As the above descriptions show, both studies are characterized by an interdisciplinary approach where we combine the concepts and methods of Artificial Intelligence with those of Social Network Analysis. However, we will not assume any particularly knowledge of these fields on the part of the reader and provide the necessary introductions to both (Chapters 2 and 3). These introductions should allow access to our work for both social scientists with an interest in electronic data and for information scientists with an interest in social-semantic applications.

Our primary goal is not to teach any of these disciplines in detail but to provide an insight for both Social and Information Scientists into the concepts and methods from outside their respective fields. We show some of the benefits that this understanding could bring in addressing complex outstanding issues that are inherently interdisciplinary in nature. Outside of the domain of Social Science we foresee further practical applications in areas where the automated, intelligent aggregation of personal electronic profiles and social networks plays an important role, mainly in digital lifestyle aggregation (connecting our fragmented online resources), but also in knowledge management, intelligence, emergency management and many others.

Our hope is also to inspire further creative experimentation toward a better understanding of both online social interaction and the nature of human knowledge. Such understanding will be indispensable in a world where the border between these once far-flung disciplines is expected to shrink rapidly through more and more socially immersive online environments such as the virtual worlds of Second Life. Only when equipped with the proper understanding will we succeed in designing systems that show true intelligence in both reasoning and social capabilities and are thus able to guide us through an ever more complex online universe.

The Author would like to acknowledge the support of the Vrije Universiteit Research School for Business Information Sciences (VUBIS) in conducting the research contained in this volume.

Bibliography

[Aberer et al., 2004] Aberer, K., Cudré-Mauroux, P., Ouksel, A. M., Catarci, T., Hacid, M.-S., Illarramendi, A., Kashyap, V., Mecella, M., Mena, E., Neuhold, E. J., Troyer, O. D., Risse, T., Scannapieco, M., Saltor, F., de Santis, L., Spaccapietra, S., Staab, S., and Studer, R. (2004). Emergent Semantics Principles and Issues. In *Database Systems for Advanced Applications 9th International Conference, DASFAA 2004*, volume 2973 of *LNCS*, pages 25–38.

[Adamic and Adar, 2005] Adamic, L. and Adar, E. (2005). How to search a social network. *Social Networks*, 27(3):"187–203".

[Adamic and Glance, 2005] Adamic, L. A. and Glance, N. (2005). The Political Blogosphere and the 2004 U.S. Election: Divided They Blog. In *Workshop on Link Discovery: Issues, Approaches and Applications (LinkKDD-2005)*.

[Adar and Adamic, 2005] Adar, E. and Adamic, L. A. (2005). Tracking Information Epidemics in Blogspace. In *Web Intelligence*, Compiegne, France.

[Adida and Birbeck, 2006] Adida, B. and Birbeck, M. (2006). RDFa Primer 1.0.

[Ahuja and Carley, 1999] Ahuja, M. K. and Carley, K. M. (1999). Network Structure in Virtual Organizations. *Organization Science*, 10(6):741–757.

[Ahuja et al., 2003] Ahuja, M. K., Galletta, D. F., and Carley, K. M. (2003). Individual Centrality and Performance in Virtual R&D Groups: An Empirical Study. *Management Science*, 49(1):21–38.

[Akkermans and Gordijn, 2006a] Akkermans, H. and Gordijn, J. (2006a). Ontology Engineering, Scientific Method, and the Research Agenda. In *Proceedings of the 15th International Conference on Knowledge Engineering and Knowledge Management (EKAW06)*.

[Akkermans and Gordijn, 2006b] Akkermans, H. and Gordijn, J. (2006b). What is This Science Called Requirements Engineering? In *14th IEEE International Requirements Engineering Conference (RE'06)*, pages 266–271, Los Alamitos, CA, USA. IEEE Computer Society.

[Albert-László Barabási and Réka Albert, 1999] Albert-László Barabási and Réka Albert (1999). Emergence of scaling in random networks. *Science*, 286:509–512.

[Anjewierden and Efimova, 2006] Anjewierden, A. and Efimova, L. (2006). Understanding weblog communities through digital traces: a framework, a tool and an example. In *International Workshop on Community Informatics (COMINF 2006)*, Montpellier, France.

[Antoniou and van Harmelen, 2004] Antoniou, G. and van Harmelen, F. (2004). *A Semantic Web Primer*. MIT Press.

[Appelquist, 2005] Appelquist, D. (2005). Enabling the mobile web through semantically-driven user experiences. In *Third Workshop on Emerging Applications for Wireless and Mobile Access (MobEA III)*.

[Baldwin, Timothy T. et al., 1997] Baldwin, Timothy T., Bedell, Michael D., and Johnson, Jonathan L. (1997). The social fabric of a team-based m.b.a. program: Network effects on student satisfaction and performance. *The Academy of Management Journal*, 40(6):1369–1397.

[Barabási et al., 2002] Barabási, A., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., and Vicsek, T. (2002). Evolution of the social network of scientific collaborations. *Physica A*, 311(3-4):590–614.

[Batagelj and Mrvar, 1998] Batagelj, V. and Mrvar, A. (1998). Pajek - Program for Large Network Analysis. *Connections*, 21(2):47–57.

[Baum et al., 2000] Baum, J. A. C., Calabrese, T., and Silverman, B. S. (2000). Don't go it alone: alliance network composition and startups' performance in Canadian biotechnology. *Strategic Management Journal*, 21:267–294.

[Bekkerman and McCallum, 2005] Bekkerman, R. and McCallum, A. (2005). Disambiguating Web Appearances of People in a Social NetworkE. In *Proceedings of the 14th International World Wide Web Conference (WWW2005)*, pages 463–470. ACM Press.

[Berners-Lee et al., 1998] Berners-Lee, T., Fielding, R., and Masinter, L. (1998). Uniform Resource Identifiers (URI): Generic Syntax.

[Berners-Lee et al., 1999] Berners-Lee, T., Fischetti, M., and Dertouzos, M. L. (1999). *Weaving the Web : The Original Design and Ultimate Destiny of the World Wide Web by its Inventor*. Harper San Francisco.

[Berners-Lee et al., 2001] Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The Semantic Web. *Scientific American*.

[Boase et al., 2006] Boase, J., Horrigan, J. B., Wellman, B., and Rainie, L. (2006). The Strength of Internet Ties. Technical report, Pew Internet & American Life Project.

[Bollegala et al., 2006] Bollegala, D., Matsuo, Y., and Ishizuka, M. (2006). Disambiguating Personal Names on the Web using Automatically Extracted Key Phrases. In *Proceedings of the 17th European Conference on Artificial Intelligence*.

[Borgatti, 1997] Borgatti, S. P. (1997). Structural Holes: Unpacking Burt's Redundancy Measures. *Connections*, 20(1):35–38.

[Borgatti et al., 2002] Borgatti, S. P., Everett, M. G., and Freeman, L. C. (2002). UCINET for Windows: Software for Social Network Analysis. Technical report, Analytic Technologies.

[Borst et al., 1997] Borst, W., Akkermans, J., and Top, J. (1997). Engineering ontologies. *International Journal of Human-Computer Studies*, 46:365–406.

[Brandes et al., 2001] Brandes, U., Eiglsperger, M., Herman, I., Himsolt, M., and Marshall, M. (2001). Graphml progress report: Structural layer proposal. In *Proceedings of the 9th International Symposium on Graph Drawing (GD '01)*, volume 2265 of *LNCS*, pages 501–512. Springer.

[Brandes et al., 2004] Brandes, U., Eiglsperger, M., and Lerner, J. (2004). GraphML Primer.

[Broekstra et al., 2002] Broekstra, J., Kampman, A., and van Harmelen, F. (2002). Sesame: An Architecture for Storing and Querying RDF and RDF Schema. In *Proceedings of the First International Semantic Web Conference (ISWC 2002)*, number 2342 in Lecture Notes in Computer Science (LNCS), pages 54–68. Springer-Verlag.

[Burt, 1995] Burt, R. S. (1995). *Structural Holes: The Social Structure of Competition*. Harvard University Press.

[Burt, 2000] Burt, R. S. (2000). The network structure of social capital. *Research in Organizational Behaviour*, 22:345–423.

[Burt, 2004] Burt, R. S. (2004). Structural Holes and Good Ideas (in press). *American Journal of Sociology*, 110(2).

[Chen, 1976] Chen, P. P. (1976). The entity-relationship model - toward a unified view of data. *ACM Trans. Database Syst.*, 1(1):9–36.

[Cicourel, 1973] Cicourel, A. V. (1973). *Cognitive Sociology*. Penguin Books, Harmondsworth, England.

[Cimiano et al., 2004] Cimiano, P., Handschuh, S., and Staab, S. (2004). Towards the Self-Annotating Web. In *Proceedings of the 13th International World Wide Web Conference*, pages 462–471, New York, USA.

[Clark, 2006] Clark, K. G. (2006). Sparql protocol for rdf.

[Coleman, 1988] Coleman, J. (1988). Social capital in the creation of human capital. *American Journal of Sociology*, 94:95–120.

[Crane, 1971] Crane, D. (1971). Transnational networks in basic science. *International Organization*, 25:585–601.

[Davies et al., 2003] Davies, J., Fensel, D., and van Harmelen, F., editors (2003). *Towards the Semantic Web: Ontology-Driven Knowledge Management*. John Wiley & Sons.

[de Sompel et al., 2006] de Sompel, H. V., Hammond, T., Neylon, E., and Weibel, S. (2006). The 'info' URI Scheme for Information Assets with Identifiers in Public Namespaces.

[deSolla Price, 1965] deSolla Price, D. J. (1965). Networks of scientific papers: The pattern of bibliographic references indicates the nature of the scientific research front. *Science*, 149(3683):510–515.

[Ding et al., 2005] Ding, L., Zhou, L., Finin, T., and Joshi, A. (2005). How the Semantic Web is Being Used: An Analysis of FOAF Documents. In *Proceedings of the 38th International Conference on System Sciences*.

[Etzioni et al., 2004] Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A.-M., Shaked, T., Soderland, S., Weld, D. S., and Yates, A. (2004). Web Scale Information Extraction in KnowItAll (Preliminary Results). In *Proceedings of the 13th International World Wide Web Conference*, pages 100–111, New York, USA.

[Fellbaum, 1998] Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

[Festinger, 1957] Festinger, L. (1957). *A theory of cognitive dissonance*. Stanford University Press, Stanford, CA.

[Fowler, 2003] Fowler, M. (2003). *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. Addison-Wesley, Third Edition edition.

[Gangemi and Mika, 2003] Gangemi, A. and Mika, P. (2003). Understanding the Semantic Web through Descriptions and Situations. In Meersman, R., Tari, Z., and et al., D. S., editors, *On The Move 2003 Conferences (OTM2003)*. Springer Verlag.

[Gargiulo and Benassi, 2000] Gargiulo, M. and Benassi, M. (2000). Trapped in Your Own Net? Network Cohesion, Structural Holes, and the Adaptation of Social Capital. *Organization Science*, 11(2):183–196.

[Gillespie, 1993] Gillespie, R. (1993). *Manufacturing Knowledge: A History of the Hawthorne Experiments*. Studies in Economic History and Policy: USA in the Twentieth Century. Cambridge University Press.

[Girvan and Newman,] Girvan, M. and Newman, M. E. J. Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA*.

[Gloor et al., 2003] Gloor, P. A., Laubacher, R., Dynes, S. B. C., and Zhao, Y. (2003). Visualization of Communication Patterns in Collaborative Innovation Networks - Analysis of Some W3C Working Groups. In *CIKM '03: Proceedings of the Twelfth International Conference on Information and Knowledge Management*, pages 56–60. ACM Press.

[Granovetter, 1973] Granovetter, M. (1973). The strength of weak ties. *American Journal of Sociology*, 78(6):1360–1380.

[Granovetter, 1992] Granovetter, M. S. (1992). Problems of explanation in economic sociology. In Nohria, N. and Eccles, R., editors, *Networks and Organizations: Structure, Form, and Action*, pages 25–56. Harvard University School Press.

[Grobelnik and Mladenic, 2002] Grobelnik, M. and Mladenic, D. (2002). Approaching Analysis of EU IST Projects Database. In *Proceedings of the International Conference on Information and Intelligent Systems (IIS-2002)*.

[Gruber, 1993] Gruber, T. R. (1993). Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In Guarino, N. and Poli, R., editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands. Kluwer Academic Publishers.

[Gruhl et al., 2004] Gruhl, D., Guha, R. V., Liben-Nowell, D., and Tomkins, A. (2004). Information diffusion through blogspace. In *Proceedings of the 13th International World Wide Web Conference*, pages 491–501, New York, USA.

[Guarino, 1998] Guarino, N. (1998). *Formal Ontology in Information Systems*. IOS Press.

[Haase et al., 2004] Haase, P., Broekstra, J., Ehrig, M., Menken, M., Mika, P., Plechawski, M., Pyszlak, P., Schnizler, B., Siebes, R., Staab, S., and Tempich, C. (2004). Bibster — a semantics-based bibliographic peer-to-peer system. In McIlraith, S. A., Plexousakis, D., and van Harmelen, F., editors, *Proceedings of the Third International Semantic Web Conference (ISWC 2004)*, pages 122–136, Hiroshima, Japan. Springer-Verlag.

[Haase et al., 2005] Haase, P., Schnizler, B., Broekstra, J., Ehrig, M., van Harmelen, F., Menken, M., Mika, P., Plechawski, M., Pyszlak, P., Siebes, R., Staab, S., and Tempich, C. (2005). Bibster — a semantics-based bibliographic peer-to-peer system. *Journal of Web Semantics*, 2(1).

[Hart et al., 2004] Hart, L., Emery, P., Colomb, B., Raymond, K., Taraporewalla, S., Chang, D., Ye, Y., Kendall, E., and Dutra, M. (2004). Owl full and uml 2.0 compared.

[Hayes, 2004] Hayes, P. (2004). Rdf semantics.

[Heimeriks et al., 2003] Heimeriks, G., Hoerlesberger, M., and van den Besselaar, P. (2003). Mapping communication and collaboration in heterogeneous research networks. *Scientometrics*, 58(2):391–413.

[Hite, 2003] Hite, J. M. (2003). Patterns of multidimensionality in embedded network ties: A typology of relational embeddedness in emerging entrepreneurial firms. *Strategic Organization!*, 1:11–52.

[Hite and Hesterly, 2001] Hite, J. M. and Hesterly, W. S. (2001). The evolution of firm networks. *Strategic Management Journal*, 22(3):275–286.

[Hof, 2006] Hof, R. D. (2006). My Virtual Life. *BusinessWeek*.

[Holland and Leinhardt, 1973] Holland, P. W. and Leinhardt, S. (1973). The Structural Implications of Measurement Error in Sociometry. *Journal of Mathematical Sociology*, 3:85–111.

[Horrocks et al., 2004] Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B., and Dean, M. (2004). Swrl: A semantic web rule language combining owl and ruleml.

[Ibarra and Andrews, 1993] Ibarra, H. and Andrews, S. B. (1993). Power, social influence and sense-making: Effects of network centrality and proximity on employee perceptions. *Administrative Science Quarterly*, 38:277–303.

[Kahney, 2003] Kahney, L. (2003). Making Friendsters in High Places. *Wired*.

[Kautz et al., 1997] Kautz, H., Selman, B., and Shah, M. (1997). The Hidden Web. *AI Magazine*, 18(2):27–36.

[Kiss et al., 1973] Kiss, G., Armstrong, C., Milroy, R., and Piper, J. (1973). An associative thesaurus of English and its computer analysis. Edinburgh University Press.

[Klein, 2004] Klein, M. (2004). *Change Management for Distributed Ontologies*. PhD thesis, Vrije Universiteit Amsterdam.

[Korotkiy and Top, 2006] Korotkiy, M. and Top, J. L. (2006). MoRe Semantic Web Applications. In *Proceedings of the End-User Aspects of the Semantic Web Workshop (UserSWeb)*.

[Krackhardt, 1990] Krackhardt, D. (1990). Assessing the Political Landscape: Structure, Cognition, and Power in Organizations. *Administrative Science Quarterly*, 35:342–369.

[Krebs, 2002] Krebs, V. (2002). Uncloaking terrorist networks. *First Monday*, 7(4).

[Kretschmer and Aguillo, 2004] Kretschmer, H. and Aguillo, I. (2004). Visibility of collaboration on the Web. *Scientometrics*, 61(3):405–426.

[Kumar et al., 2003] Kumar, R., Novak, J., Raghavan, P., and Tomkins, A. (2003). On the Bursty Evolution of Blogspace. In *Proceedings of the 12th International World Wide Web Conference*.

[Lemmens, 2003] Lemmens, C. (2003). *Network Dynamics and Innovation*. PhD thesis, Technical University of Eindhoven (TUE).

[Manola and Miller, 2004] Manola, F. and Miller, E. (2004). Rdf primer.

[Marsden and Campbell, 1984] Marsden, P. V. and Campbell, K. E. (1984). Measuring tie strength. *Social Forces*, 63(2):482–501.

[Masolo et al., 2004] Masolo, C., Vieu, L., Bottazzi, E., Catenacci, C., Ferrario, R., Gangemi, A., and Guarino, N. (2004). Social roles and their descriptions. In *Proceedings of the Ninth International Conference on the Principles of Knowledge Representation and Reasoning*. AAAI Press.

[Matsuo et al., 2006] Matsuo, Y., Hamasaki, M., Takeda, H., Mori, J., Bollegara, D., Nakamura, Y., Nishimura, T., Hasida, K., and Ishizuka, M. (2006). Spinning Multiple Social Networks for Semantic Web. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence (AAAI2006)*.

[Mayo, 1933] Mayo, E. (1933). *The Human Problems of an Industrial Civilization*. Macmillan, Cambridge, MA, USA.

[McGuinness and van Harmelen, 2004] McGuinness, D. L. and van Harmelen, F. (2004). OWL Web Ontology Language Overview. Technical report, World Wide Web Consortium (W3C).

[McPherson et al., 2006] McPherson, M., Smith-Lovin, L., and Brashears, M. E. (June 2006). Social isolation in america: Changes in core discussion networks over two decades. *American Sociological Review*, 71:353–375.

[Mehra et al., 1998] Mehra, A., Kilduff, M., and Brass, D. J. (1998). At the margins: A distinctiveness approach to the social identity and social networks of underrepresented groups. *Academy of Management Journal*, 41(4):441–452.

[Mehra et al., 2001] Mehra, A., Kilduff, M., and Brass, D. J. (2001). The Social Networks of High and Low Self-monitors: Implications for Workplace Performance. *Administrative Science Quarterly*, 46(2).

[Mika, 2002] Mika, P. (2002). Integrating Ontology Storage and Ontology-based Applications Through Client-side Query and Transformations. In *Proceedings of Evaluation of Ontology-based Tools (EON2002) workshop at EKAW2002, Siguenza, Spain*.

[Mika, 2004] Mika, P. (2004). Social Networks and the Semantic Web: An Experiment in Online Social Network Analysis. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, Beijing, China.

[Mika, 2005a] Mika, P. (2005a). Flink: Semantic Web Technology for the Extraction and Analysis of Social Networks. *Journal of Web Semantics*, 3(2).

[Mika, 2005b] Mika, P. (2005b). Social Networks and the Semantic Web: The Next Challenge. *IEEE Intelligent Systems*, 20(1).

[Mika and Akkermans, 2004] Mika, P. and Akkermans, H. (2004). Towards a New Synthesis of Ontology Technology and Knowledge Management. *Knowledge Engineering Review*, 19(4):317–345.

[Mika et al., 2004] Mika, P., Oberle, D., Gangemi, A., and Sabou, M. (2004). Foundations for Service Ontologies: Aligning OWL-S to DOLCE. In *Proceedings of the 13th International World Wide Web Conference (WWW2004)*. ACM Press.

[Milgram, 1967] Milgram, S. (1967). The Small World Problem. *Psychology Today*, 1(1):61–67.

[Mori et al., 2004] Mori, J., Matsuo, Y., Ishizuka, M., and Faltings, B. (2004). Keyword Extraction from the Web for FOAF Metadata. In *Proceedings of the 1st Workshop on Friend of a Friend, Social Networking and the (Semantic) Web*.

[Mutschke and Haase, 2001] Mutschke, P. and Haase, A. Q. (2001). Collaboration and cognitive structures in social science research fields. *Scientometrics*, 52(3).

[Nahapiet and Ghoshal, 1998] Nahapiet, J. and Ghoshal, S. (1998). Social Capital, Intellectual Capital, and the Organizational Advantage. *The Academy of Management Review*, 23(2):242–266.

[Otte and Rousseau, 2002] Otte, E. and Rousseau, R. (2002). Social network analysis: a powerful strategy, also for the information science. *Journal of Information Science*, 28(6):441–453.

[Paolillo et al., 2005] Paolillo, J. C., Mercure, S., and Wright, E. (2005). The Social Semantics of LiveJournal FOAF: Structure and Change from 2004 to 2005. In *Workshop on Semantic Network Analysis (SNA'05)*.

[Paolillo and Wright, 2004] Paolillo, J. C. and Wright, E. (2004). The Challenges of FOAF Characterization. In *Proceedings of the 1st Workshop on Friend of a Friend, Social Networking and the (Semantic) Web*.

[Petke and King, 1999] Petke, R. and King, I. (1999). Registration Procedures for URL Scheme Names.

[Portwin and Parvatikar, 2006] Portwin, K. and Parvatikar, P. (2006). Building and managing a massive triple store: An experience report. In *Proceedings of XTech 2006*, Amsterdam, The Netherlands.

[Powers, 2003] Powers, S. (2003). *Practical RDF*. O'Reilly Media.

[Prud'hommeaux and Seaborne, 2006] Prud'hommeaux, E. and Seaborne, A. (2006). SPARQL Query Language for RDF.

[Putnam, 1993] Putnam, R. D. (1993). The Prosperous Community: Social Capital and Public Life. *The American Prospect*, 4(13):35–42.

[Quan and Karger, 2004] Quan, D. and Karger, D. R. (2004). How to Make a Semantic Web Browser. In *Proceedings of the 13th International World Wide Web Conference*, pages 255–265, New York, USA.

[Reagans and McEvily, 2003] Reagans, R. and McEvily, B. (2003). Network Structure and Knowledge Transfer: The Effects of Cohesion and Range. *Administrative Science Quarterly*, 48(2):240–267.

[Rector et al., 2004] Rector, A., Drummond, N., Horridge, M., Rogers, J., Knublauch, H., Stevens, R., Wang, H., and Wroe, C. (2004). Owl pizzas: Practical experience of teaching owl-dl: Common errors and common patterns. In *14th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2004)*.

[Robson, 2002] Robson, C. (2002). *Real World Research: A Resource for Social Scientists and Practitioner-Researchers*. Blackwell Publishing, second edition edition.

[Ruan, 1998] Ruan, D. (1998). The content of the General Social Survey discussion networks: an exploration of General Social Survey discussion name generatornext term in a previous term-Chinese context. *Social Networks*, 20(3):247–264.

[Salton, 1989] Salton, G. (1989). *Automatic text processing*. Addison-Wesley, Reading, MA.

[Scott, 2000] Scott, J. P. (2000). *Social Network Analysis: A Handbook*. Sage Publications, 2nd edition.

[Smith and Welty, 2001] Smith, B. and Welty, C. (2001). Ontology: Towards a new synthesis. In *Formal Ontology in Information Systems*, pages iii–x, Ongunquit, Maine. ACM Press.

[Smith et al., 2004] Smith, M. K., Welty, C., and McGuinness, D. L. (2004). OWL Web Ontology Language Guide. Technical report, World Wide Web Consortium (W3C).

[ter Horst, 2005] ter Horst, H. (2005). Combining rdf and part of owl with rules: Semantics, decidability, complexity. In Gil, Y., Motta, E., Benjamins, V. R., and Musen, M. A., editors, *Proceedings of the Fourth International Semantic Web Conference (ISWC 2005)*, volume 3729 of *LNCS*, pages 668–684, Galway, Ireland. Springer-Verlag.

[Tyler et al., 2003] Tyler, J. R., Wilkinson, D. M., and Huberman, B. A. (2003). Email as spectroscopy: automated discovery of community structure within organizations. In *International Conference on Communities and Technologies*, pages 81–96, Deventer, The Netherlands. Kluwer, B.V.

[van Atteveldt et al., 2006] van Atteveldt, W., Kleinnijenhuis, J., Oegema, D., and Schlobach, S. (2006). Knowledge Representation of Social and Cognitive Networks. In *Procedings of the Social Networks Analysis workshop of the 3rd European Semantic Web Conference (ESWC06)*.

[van de Bunt, 1999] van de Bunt, G. (1999). *Friends by Choice; An Actor-Oriented Statistical Network Model for Friendship Networks through Time*. PhD thesis, Rijksuniversiteit Groningen.

[van Rijsbergen, 1979] van Rijsbergen, C. (1979). *Information Retrieval*. Butterworth-Heinemann, London, 2nd edition edition.

[Volz et al., 2003] Volz, R., Oberle, D., Staab, S., and Studer, R. (2003). Ontolift prototype. WonderWeb Deliverable 11.

[Warner and Lunt, 1941] Warner, W. L. and Lunt, P. (1941). *The Social Life of a Modern Community*. Yale University Press, New Haven, CT, USA.

[Wasserman et al., 1994] Wasserman, S., Faust, K., Iacobucci, D., and Granovetter, M. (1994). *Social Network Analysis: Methods and Applications*. Cambridge University Press.

[Watts, 1999] Watts, D. J. (1999). Networks, dynamics and the small-world phenomenon. *American Journal of Sociology*, 105(2):493–527.

[Watts and Strogatz, 1998] Watts, D. J. and Strogatz, S. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393:440–442.

[Wellman et al., 1996] Wellman, B., Salaff, J., Dimitrova, D., Garton, L., Gulia, M., and Haythornthwaite, C. (1996). Computer Networks as Social Networks: Collaborative Work, Telework, and Virtual Community. *Annual Review of Sociology*, 22:213–238.

[Wu et al., 2004] Wu, F., Huberman, B. A., Adamic, L. A., and Tyler, J. R. (2004). Information flow in social groups. *Physica A*, 337:327–335.

[Zuckerman and Reagans, 2001] Zuckerman, E. W. and Reagans, R. E. (2001). Networks, Diversity, and Performance: The Social Capital of Corporate R&D Teams. *Organization Science*, 12(4):502–517.

SIKS Dissertation Series

1998

1998-1 Johan van den Akker (CWI)
DEGAS - An Active, Temporal Database of Autonomous Objects

1998-2 Floris Wiesman (UM)
Information Retrieval by Graphically Browsing Meta-Information

1998-3 Ans Steuten (TUD)
A Contribution to the Linguistic Analysis of Business Conversations within the Language/Action Perspective

1998-4 Dennis Breuker (UM)
Memory versus Search in Games

1998-5 E.W.Oskamp (RUL)
Computerondersteuning bij Straftoemeting

1999

1999-1 Mark Sloof (VU)
Physiology of Quality Change Modelling; Automated modelling of Quality Change of Agricultural Products

1999-2 Rob Potharst (EUR)
Classification using decision trees and neural nets

1999-3 Don Beal (UM)
The Nature of Minimax Search

1999-4 Jacques Penders (UM)
The practical Art of Moving Physical Objects

1999-5 Aldo de Moor (KUB) *Empowering Communities: A Method for the Legitimate User-Driven Specification of Network Information Systems*

1999-6 Niek J.E. Wijngaards (VU)
Re-design of compositional systems

1999-7 David Spelt (UT)
Verification support for object database design

1999-8 Jacques H.J. Lenting (UM)
Informed Gambling: Conception and Analysis of a Multi-Agent Mechanism for Discrete Reallocation

2000

2000-1 Frank Niessink (VU)
Perspectives on Improving Software Maintenance

2000-2 Koen Holtman (TUE) *Prototyping of CMS Storage Management*

2000-3 Carolien M.T. Metselaar (UvA)
Sociaal-organisatorische gevolgen van kennistechnologie; een procesbenadering en actorperspectie

2000-4 Geert de Haan (VU)
ETAG, A Formal Model of Competence Knowledge for User Interface Design

2000-5 Ruud van der Pol (UM)
Knowledge-based Query Formulation in Information Retrieval

2000-6 Rogier van Eijk (UU)
Programming Languages for Agent Communication

2000-7 Niels Peek (UU)
Decision-theoretic Planning of Clinical Patient Management

2000-8 Veerle Coupé (EUR)
Sensitivity Analyis of Decision-Theoretic Networks

2000-9 Florian Waas (CWI)
Principles of Probabilistic Query Optimization

2000-10 Niels Nes (CWI)
Image Database Management System Design Considerations, Algorithms and Architecture

2000-11 Jonas Karlsson (CWI)
Scalable Distributed Data Structures for Database Management

2001

2001-1 Silja Renooij (UU)
Qualitative Approaches to Quantifying Probabilistic Networks

2001-2 Koen Hindriks (UU)
Agent Programming Languages: Programming with Mental Models

2001-3 Maarten van Someren (UvA)
Learning as problem solving

2001-4 Evgueni Smirnov (UM)
Conjunctive and Disjunctive Version Spaces with Instance-Based Boundary Sets

2001-5 Jacco van Ossenbruggen (VU)
Processing Structured Hypermedia: A Matter of Style

2001-6 Martijn van Welie (VU)
Task-based User Interface Design

2001-7 Bastiaan Schonhage (VU)
Diva: Architectural Perspectives on Information Visualization

2001-8 Pascal van Eck (VU)
A Compositional Semantic Structure for Multi-Agent Systems Dynamics

2001-9 Pieter Jan 't Hoen (RUL)
Towards Distributed Development of Large Object-Oriented Models, Views of Packages as Classes

2001-10 Maarten Sierhuis (UvA)
Modeling and Simulating Work Practice BRAHMS: a multiagent modeling and simulation language for work practice analysis and design

2001-11 Tom M. van Engers (VU)
Knowledge Management: The Role of Mental Models in Business Systems Design

2002

2002-01 Nico Lassing (VU)
Architecture-Level Modifiability Analysis

2002-02 Roelof van Zwol (UT)
Modelling and searching web-based document collections

2002-03 Henk Ernst Blok (UT)
Database Optimization Aspects for Information Retrieval

2002-04 Juan Roberto Castelo Valdueza (UU)
The Discrete Acyclic Digraph Markov Model in Data Mining

2002-05 Radu Serban (VU)
The Private Cyberspace Modeling Electronic Environments inhabited by Privacy-concerned Agents

2002-06 Laurens Mommers (UL)
Applied legal epistemology; Building a knowledge-based ontology of the legal domain

2002-07 Peter Boncz (CWI)
Monet: A Next-Generation DBMS Kernel For Query-Intensive Applications

2002-08 Jaap Gordijn (VU)
Value Based Requirements Engineering: Exploring Innovative E-Commerce Ideas

2002-09 Willem-Jan van den Heuvel (KUB)
Integrating Modern Business Applications with Objectified Legacy Systems

2002-10 Brian Sheppard (UM)
Towards Perfect Play of Scrabble

2002-11 Wouter C.A. Wijngaards (VU)
Agent Based Modelling of Dynamics: Biological and Organisational Applications

2002-12 Albrecht Schmidt (UvA)
Processing XML in Database Systems

2002-13 Hongjing Wu (TUE)
A Reference Architecture for Adaptive Hypermedia Applications

2002-14 Wieke de Vries (UU)
Agent Interaction: Abstract Approaches to Modelling, Programming and Verifying Multi-Agent Systems

2002-15 Rik Eshuis (UT)
Semantics and Verification of UML Activity Diagrams for Workflow Modelling

2002-16 Pieter van Langen (VU)
The Anatomy of Design: Foundations, Models and Applications

2002-17 Stefan Manegold (UvA)
Understanding, Modeling, and Improving Main-Memory Database Performance

2003

2003-01 Heiner Stuckenschmidt (VU)
Ontology-Based Information Sharing in Weakly Structured Environments

2003-02 Jan Broersen (VU)
Modal Action Logics for Reasoning About Reactive Systems

2003-03 Martijn Schuemie (TUD)
Human-Computer Interaction and Presence in Virtual Reality Exposure Therapy

2003-04 Milan Petkovic (UT)
Content-Based Video Retrieval Supported by Database Technology

2003-05 Jos Lehmann (UvA)
Causation in Artificial Intelligence and Law - A modelling approach

2003-06 Boris van Schooten (UT)
Development and specification of virtual environments

2003-07 Machiel Jansen (UvA)
Formal Explorations of Knowledge Intensive Tasks

2003-08 Yongping Ran (UM)
Repair Based Scheduling

2003-09 Rens Kortmann (UM)
The resolution of visually guided behaviour

2003-10 Andreas Lincke (UvT)
Electronic Business Negotiation: Some experimental studies on the interaction between medium, innovation context and culture

2003-11 Simon Keizer (UT)
Reasoning under Uncertainty in Natural Language Dialogue using Bayesian Networks

2003-12 Roeland Ordelman (UT)
Dutch speech recognition in multimedia information retrieval

2003-13 Jeroen Donkers (UM)
Nosce Hostem - Searching with Opponent Models

2003-14 Stijn Hoppenbrouwers (KUN)
Freezing Language: Conceptualisation Processes across ICT-Supported Organisations

2003-15 Mathijs de Weerdt (TUD)
Plan Merging in Multi-Agent Systems

2003-16 Menzo Windhouwer (CWI)
Feature Grammar Systems - Incremental Maintenance of Indexes to Digital Media Warehouses

2003-17 David Jansen (UT)
Extensions of Statecharts with Probability, Time, and Stochastic Timing

2003-18 Levente Kocsis (UM)
Learning Search Decisions

2004

2004-01 Virginia Dignum (UU)
A Model for Organizational Interaction: Based on Agents, Founded in Logic

2004-02 Lai Xu (UvT)
Monitoring Multi-party Contracts for E-business

2004-03 Perry Groot (VU)
A Theoretical and Empirical Analysis of Approximation in Symbolic Problem Solving

2004-04 Chris van Aart (UvA)
Organizational Principles for Multi-Agent Architectures

2004-05 Viara Popova (EUR)
Knowledge discovery and monotonicity

2004-06 Bart-Jan Hommes (TUD)
The Evaluation of Business Process Modeling Techniques

2004-07 Elise Boltjes (UM)
Voorbeeldig onderwijs; voorbeeldgestuurd onderwijs, een opstap naar abstract denken, vooral voor meisjes

2004-08 Joop Verbeek (UM)
Politie en de Nieuwe Internationale Informatiemarkt, Grensregionale politiële gegevensuitwisseling en digitale expertise

2004-09 Martin Caminada (VU)
For the Sake of the Argument; explorations into argument-based reasoning

2004-10 Suzanne Kabel (UvA)
Knowledge-rich indexing of learning-objects

2004-11 Michel Klein (VU)
Change Management for Distributed Ontologies

2004-12 The Duy Bui (UT)
Creating emotions and facial expressions for embodied agents

2004-13 Wojciech Jamroga (UT)
Using Multiple Models of Reality: On Agents who Know how to Play

2004-14 Paul Harrenstein (UU)
Logic in Conflict. Logical Explorations in Strategic Equilibrium

2004-15 Arno Knobbe (UU)
Multi-Relational Data Mining

2004-16 Federico Divina (VU)
Hybrid Genetic Relational Search for Inductive Learning

2004-17 Mark Winands (UM)
Informed Search in Complex Games

2004-18 Vania Bessa Machado (UvA)
Supporting the Construction of Qualitative Knowledge Models

2004-19 Thijs Westerveld (UT)
Using generative probabilistic models for multimedia retrieval

2004-20 Madelon Evers (Nyenrode)
Learning from Design: facilitating multidisciplinary design teams

2005

2005-01 Floor Verdenius (UVA)
Methodological Aspects of Designing Induction-Based Applications

2005-02 Erik van der Werf (UM))
AI techniques for the game of Go

2005-03 Franc Grootjen (RUN)
A Pragmatic Approach to the Conceptualisation of Language

2005-04 Nirvana Meratnia (UT)
Towards Database Support for Moving Object data

2005-05 Gabriel Infante-Lopez (UVA)
Two-Level Probabilistic Grammars for Natural Language Parsing

2005-06 Pieter Spronck (UM)
Adaptive Game AI

2005-07 Flavius Frasincar (TUE)
Hypermedia Presentation Generation for Semantic Web Information Systems

2005-08 Richard Vdovjak (TUE)
A Model-driven Approach for Building Distributed Ontology-based Web Applications

2005-09 Jeen Broekstra (VU)
Storage, Querying and Inferencing for Semantic Web Languages

2005-10 Anders Bouwer (UVA)
Explaining Behaviour: Using Qualitative Simulation in Interactive Learning Environments

2005-11 Elth Ogston (VU)
Agent Based Matchmaking and Clustering - A Decentralized Approach to Search

2005-12 Csaba Boer (EUR)
Distributed Simulation in Industry

2005-13 Fred Hamburg (UL)
Een Computermodel voor het Ondersteunen van Euthanasiebeslissingen

2005-14 Borys Omelayenko (VU)
Web-Service configuration on the Semantic Web; Exploring how semantics meets pragmatics

2005-15 Tibor Bosse (VU)
Analysis of the Dynamics of Cognitive Processes

2005-16 Joris Graaumans (UU)
Usability of XML Query Languages

2005-17 Boris Shishkov (TUD)
Software Specification Based on Reusable Business Components

2005-18 Danielle Sent (UU)
Test-selection strategies for probabilistic networks

2005-19 Michel van Dartel (UM)
Situated Representation

2005-20 Cristina Coteanu (UL)
Cyber Consumer Law, State of the Art and Perspectives

2005-21 Wijnand Derks (UT)
Improving Concurrency and Recovery in Database Systems by Exploiting Application Semantics

2006

2006-01 Samuil Angelov (TUE)
Foundations of B2B Electronic Contracting

2006-02 Cristina Chisalita (VU)
Contextual issues in the design and use of information technology in organizations

2006-03 Noor Christoph (UVA)
The role of metacognitive skills in learning to solve problems

2006-04 Marta Sabou (VU)
Building Web Service Ontologies

2006-05 Cees Pierik (UU)
Validation Techniques for Object-Oriented Proof Outlines

2006-06 Ziv Baida (VU)
Software-aided Service Bundling - Intelligent Methods & Tools for Graphical Service Modeling

2006-07 Marko Smiljanic (UT)
XML schema matching – balancing efficiency and effectiveness by means of clustering

2006-08 Eelco Herder (UT)
Forward, Back and Home Again - Analyzing User Behavior on the Web

2006-09 Mohamed Wahdan (UM)
Automatic Formulation of the Auditor's Opinion

2006-10 Ronny Siebes (VU)
Semantic Routing in Peer-to-Peer Systems

2006-11 Joeri van Ruth (UT)
Flattening Queries over Nested Data Types

2006-12 Bert Bongers (VU)
Interactivation - Towards an e-cology of people, our technological environment, and the arts

2006-13 Henk-Jan Lebbink (UU)
Dialogue and Decision Games for Information Exchanging Agents

2006-14 Johan Hoorn (VU)
Software Requirements: Update, Upgrade, Redesign - towards a Theory of Requirements Change

2006-15 Rainer Malik (UU)
CONAN: Text Mining in the Biomedical Domain

2006-16 Carsten Riggelsen (UU)
Approximation Methods for Efficient Learning of Bayesian Networks

2006-17 Stacey Nagata (UU)
User Assistance for Multitasking with Interruptions on a Mobile Device

2006-18 Valentin Zhizhkun (UVA)
Graph transformation for Natural Language Processing

2006-19 Birna van Riemsdijk (UU)
Cognitive Agent Programming: A Semantic Approach

2006-20 Marina Velikova (UvT)
Monotone models for prediction in data mining

2006-21 Bas van Gils (RUN)
Aptness on the Web

2006-22 Paul de Vrieze (RUN)
Fundaments of Adaptive Personalisation

2006-23 Ion Juvina (UU)
Development of Cognitive Model for Navigating on the Web

2006-24 Laura Hollink (VU)
Semantic Annotation for Retrieval of Visual Resources

2006-25 Madalina Drugan (UU)
Conditional log-likelihood MDL and Evolutionary MCMC

2006-26 Vojkan Mihajlovic (UT)
Score Region Algebra: A Flexible Framework for Structured Information Retrieval

2006-27 Stefano Bocconi (CWI)
Vox Populi: generating video documentaries from semantically annotated media repositories

2006-28 Borkur Sigurbjornsson (UVA) *Focused Information Access using XML Element Retrieval*

2007

2007-01 Kees Leune (UvT)
Access Control and Service-Oriented Architectures

2007-02 Wouter Teepe (RUG)
Reconciling Information Exchange and Confidentiality: A Formal Approach