Università di Pisa

Dipartimento di Ingegneria dell’Informazione
Dottorato di Ricerca in
Ingegneria dell’Informazione

Ph.D. Thesis

Sharing Semantic Resources

Ing. Maurizio Tesconi

Supervisor	Supervisor
Prof. Francesco Marcelloni	Prof. Andrea Tomasi

Supervisor
Dott. Andrea Marchetti

2007

Abstract

The Semantic Web is an extension of the current Web in which information, so far created for human consumption, becomes machine readable, "enabling computers and people to work in cooperation". To turn into reality this vision several challenges are still open among which the most important is to share meaning formally represented with ontologies or more generally with semantic resources. This Semantic Web long-term goal has many convergences with the activities in the field of Human Language Technology and in particular in the development of Natural Language Processing applications where there is a great need of multilingual lexical resources. For instance, one of the most important lexical resources, WordNet, is also commonly regarded and used as an ontology. Nowadays, another important phenomenon is represented by the explosion of social collaboration, and Wikipedia, the largest encyclopedia in the world, is object of research as an up to date omni comprehensive semantic resource. The main topic of this thesis is the management and exploitation of semantic resources in a collaborative way, trying to use the already available resources as Wikipedia and Wordnet. This work presents a general environment able to turn into reality the vision of shared and distributed semantic resources and describes a distributed three-layer architecture to enable a rapid prototyping of cooperative applications for developing semantic resources.

Acknowledgments

This thesis has been prepared while I was working at the Institute of Informatics and Telematics (IIT) of the National Research Council (CNR) of Pisa. I would like to thank my advisors. I gratefully acknowledge Andrea Marchetti for his support and helpful comments during the preparation of this thesis. I would also like to thank Oreste Signore for providing valuable feedback. Many thanks go to the members of the Security Research Group for contributing to this thesis. In particular, I would like to thank Francesco Ronzano for many fruitful discussions and valuable collaboration. I would also like to thank Salvatore Minutoli who carefully reviewed the entire first draft. Furthermore, I would like to thank Marco Rosella for his contribution in the implementation of a first version of the SemKey prototype presented in this thesis, Claudia Soria for providing useful feedback on the thesis, Fabrizio Falchi for many interesting discussions. Special thanks go to my parents, Mariella and Giuseppe, who always supported me throughout my studies. Last but not least, I would like to dedicate this work to Letizia for her loving support and patience during the last years.

Contents

II Management of Semantic Resources 33

List of Tables

List of Figures

To Lety.

Introduction

Nowadays when we consider the Web we think about a lot of linked data regarding a huge number of different knowledge domains. All this available data has a great *semantic* informative value, but many times it is not possible to take real advantage of it, due to its different and often chaotic organization and structure. Integration and interoperability between multiple different sources of information are usually difficult and frequently the adopted solutions are very expensive to realize and tailored to specific situations.

The Semantic Web is an attempt to gradually solve this problems adding richer and more structured descriptive content to Web data. The greatest part of current Web content is built exclusively for humans, meaning that most of Web information is available as textual content, opportunely structured in order to be rendered by Web browsers. The Semantic Web aims at going beyond the textual level of content delivery. As stated by Tim Berners-Lee, its major deviser in 2001 [BLTO01], the Semantic Web "is an extension of the current Web in which information is given a well defined meaning, better enabling computers and people to work in cooperation".

In order to make the Semantic Web a reality, the main problem we need to solve is *ontology availability* [BC02]. A natural convergence thus exists between the Semantic Web long-term goals and some of the core activities in the field of Human Language Technology (HLT). In the Semantic Web, content is annotated with respect to particular ontologies, which provide the definition of the basic vocabulary and semantics of the annotations. Generally, ontologies appear as key ingredients in knowledge management and content based systems, with applications relating to document search and categorization, e-commerce, agent-to-agent communication, etc. In HLT, the task of providing the basic semantic description of words is entrusted to computational lexicons, which therefore represent critical information sources for most NLP systems. The availability of large-scale repositories of lexical information is in fact an essential precondition for HLT to be able to tackle the full complexity of multilingual text processing. Ontologies also represent an important bridge between knowledge representation and computational lexical semantics, and currently form a point of convergence with semantic lexicons. In fact, they are widely used (together with lexicons) to represent the lexical content of words, and appear to have a crucial role in different natural language processing (NLP) tasks, such as content-based tagging, word sense disambiguation, multilingual transfer, etc. Besides, one of the most widely used lexical resources, WordNet [Wne], is also commonly regarded and used as an ontology, as further evidence of the commonalities existing between computational lexicons and ontologies [Gua98] [OGGM]. Exploring the world of *semantic resources* and analyzing the synergies between ontology design and computational lexicon becomes a fundamental step. The main topic of this thesis is the management and exploitation of semantic resources or, to put it better the study of new approach to develop and use semantic resource in a collaborative way.

I.1 Management of Semantic Resources

Today we can observe a proliferation of multilingual, Web-based, lexical resources. Moreover, market calls for new types of semantic resources, rapidly built and easy tailored, exploiting the richness of existing resources. This scenario no longer leaves space to static, closed, and locally managed repositories of semantic information; instead, it calls for an environment where semantic resources can be shared are reusable, and are openly customizable. At the same time, as the history of the web teaches, it would be a mistake to create a central repository containing all the shared resources because of the difficulties to manage it. Distribution of resources thus becomes a central concept: the idea consists in moving towards distributed general-purpose resources, based on open content interoperability standards, and made accessible to users via web-services technologies. There is another, deeper argument in favor of distributed semantic resources: language resources, lexicons included, are inherently distributed because of the diversity of languages distributed over the world, that makes it impossible to have one single centralized repository of resources. In this way, each resource is developed and maintained in its natural environment. This new type of semantic resources can still be stored locally, but its maintenance and exploitation can be a matter of agents being choreographed to act over them. A possible solution of this problem comes from collaborative tools such as applications for workflow orchestration. Admittedly, this is a long-term scenario requiring the contribution of many different actors and initiatives. The first prerequisite for this project fulfillment is to ensure true interoperability among semantic resources, a goal that is long being addressed to by the standardization community. Although the paradigm of distributed and interoperable semantic resources has largely been discussed and invoked, very little has been made for the development of new methods and techniques for its practical achievement. In order to overcome limits due to singular resources and reach a common knowledge platform, a change in the very basic assumptions on the design, creation, maintenance and distribution of knowledge resources is needed and for this purpose some very interesting suggestions come from the web. In particular, the emerging operational and theoretical paradigm based on the notions of cooperation, collaboration and social knowledge determination seems to offer a way to rethink the entire strategy of creation and management of semantic resources. Collaboration is what subtends the practice of groups asynchronously producing works together through individual contributions in the so-called collaborative authoring.

In this thesis we want to present a general architecture for the management and exploitation of semantic resources. For 'management' we intend not only create and edit resources but also integrate existing ones by using approaches of cross-fertilization. In chapter 2 we present a general environment able to turn into reality the vision of shared and distributed semantic repositories. We designed a distributed three-layer architecture to enable a rapid prototyping of cooperative applications for developing semantic resources. The higher layer was built on XFlow[MTM05], a framework cooperative management of XML resources that has been developed during the first part of my research activity in the Institute of Informatics and Telematics (IIT) of the National Research Council (CNR) of Pisa. (see Chapter 3). The cooperative layer or LeXFlow[TMB$^+$06] (see chapter 4) is intended as an overall environment where all the modules implemented in the lower layers can be integrated in a comprehensive workflow of human and software agents. The middle layer hosts some applications that exploit the semantic shared repositories. The so-called MultiWordNet Service (MWS)[STM$^+$06] allows to mutually enrich wordnets in a distributed environment (see Chapter 5). In the same layer there is the SemKey prototype, another application that exploit the shared ontology for disambiguating keywords. Other and more advanced Natural Language Processing (NLP) applications can be developed by exploiting the availability of the repositories. The lower layer consists of a sort of a grid of local services realized as a virtual repository of XML databases residing at different locations and accessible through web services. Basic software services are also necessary, such as an UDDI server for the registration of the local wordnets and web services dedicated to the coherent management of the different versions of WordNet the databases referred by databases. In this thesis we concentrate on the description of cooperative layer and the application layer.

I.2 Exploiting Semantic Resources

Collaborative tagging is a new content sharing and organizational trend, mainly diffused over the Web, which has attracted increasing attention in last few years. It refers to the process by which many users add metadata in the form of keywords to shared content. Today many different collaborative tagging systems are available on the Web, enabling users to add descriptive keywords to different types of Internet resources (web pages, photos, videos, etc.). The great number of advantages offered by the availability of collaboratively tagged resources in terms of their organization and shared information is underlined by their growing adoption, also in non-technical communities of users. In spite of this, by analyzing the current structure and usage patterns of collaborative tagging systems, we can discover many important aspects which still need to be improved in order to bring tagging systems to their full potential. In particular, problems related to synonymy, polysemy, different lexical forms, different spellings and misspelling errors, but also the lack of accurancy caused by different levels of precision and distinct kinds of tag-to-resource association represent a great limit, causing inconsistencies among the terms used in the tagging process and thus reducing the efficiency of content search and the effectiveness of the tag space structuring and organization. This kind of problems is mainly caused by the lack of semantic information in the tagging process. Examining the different causes of inconsistencies and loss of precision in tag-space based searches, we can infer that most of them may be solved or substantially reduced bringing semantics to collaborative tagging systems. Each tag should not represent just a simple sequence of characters, but should be defined by specifying its meaning. When a user decides to tag resources, describing them by means of one or more keywords, he must be able to disambiguate each tag, defining its semantics or better pointing out its contextualized meaning. Moreover, we introduce properties to link concepts to a specific resource; this process will be referred to as *semantic collaborative tagging*. In this way, the outcome of semantic tagging activity consists of *producing a set of unambiguous assertions on resources*. Each of them could represent statements about the topic or kind of resource or concern the user opinion about the web resources. In the last part of this thesis (Chapter 6) we analyze how semantic resources can be exploited in the context of semantic tagging.

Part I

Semantic Resources

Chapter 1

Computational lexicons and Semantic Web

Abstract

The vision of the Semantic Web is to turn the World Wide Web into a machine-understandable knowledge base, thereby allowing agents and applications to access a variety of heterogeneous resources by processing and integrating their contents. In order to make the Semantic Web a reality, it is therefore necessary to solve the problem of ontology availability[BC02]. The Semantic Web long-term goal has many convergences with the activities in the field of Human Language Technology and in particular in the development of Natural Language Processing applications.

The main topic of this chapter is the analyzing the synergies between ontology design and computational lexicons. Section 1.2 analyses the more relevant knowledge structures for representing semantic information. Section 1.3 discusses the Semantic Web vision, Section 1.4 describes the world of computational lexicons and Section 1.5 shows the structure and features of Wordnet. Finally, in Section 1.6 we discuss about the new social trend that allows the building of new open, free content, general-purpose lexical resources.

1.1 Ontology design and computational lexicon

A natural convergence thus exists between the Semantic Web long-term goals and some of the core activities in the field of Human Language Technology (HLT). In the Semantic Web, content is annotated with respect to particular ontologies, which provide the definition of the basic vocabulary and semantics of the annotations. More in general, ontologies appear as key ingredients in knowledge management and content based systems, with applications ranging from document search and categorization, e-commerce, agent-to-agent communication, etc. In HLT, the task of providing the basic semantic description of words is entrusted to computational lexicons, which therefore represent critical information sources for most Natural Language Processing (NLP) systems. The availability of large-scale repositories of lexical information is in fact an essential precondition for HLT to be able to tackle the full complexity of multilingual text processing.

Ontologies also represent an important bridge between knowledge representation and computational lexical semantics, and currently form a *continuum* with semantic lexicons. In fact, they are widely used (together with lexicons) to represent the lexical content of words, and appear to have a crucial role in different NLP tasks, such as content-based tagging, word sense disambiguation, multilingual transfer, etc. Besides, one of the most widely used lexical resources, WordNet [Wne], is also commonly regarded and used as an ontology, as further evidence of the commonalities existing between computational lexicons and ontologies [Gua98] [OGGM].

1.2 Knowledge Organization

The importance of knowledge based systems (KBS) has been analysed in information sciences since the 1980s, mostly in relation to automatic indexing (natural language processing) and information retrieval [Cro95]. The term knowledge organization systems (KOS) is intended to encompass all types of schemes for organizing information and promoting knowledge management. Knowledge organization systems also include highly structured vocabularies, such as thesauri, and less traditional schemes, such as semantic networks and ontologies. Because KOS are mechanisms for organizing information, they are at the heart of every library, museum, and archive. In knowledge management abuse of terminology for systems of organizing knowledge is rampant. In this section is given an overview of a variety of knowledge structures. There are many more of them but here we only describe a selection illustrating the main different typologies. A possible organization of these knowledge structures as shown in Figure 1.1.

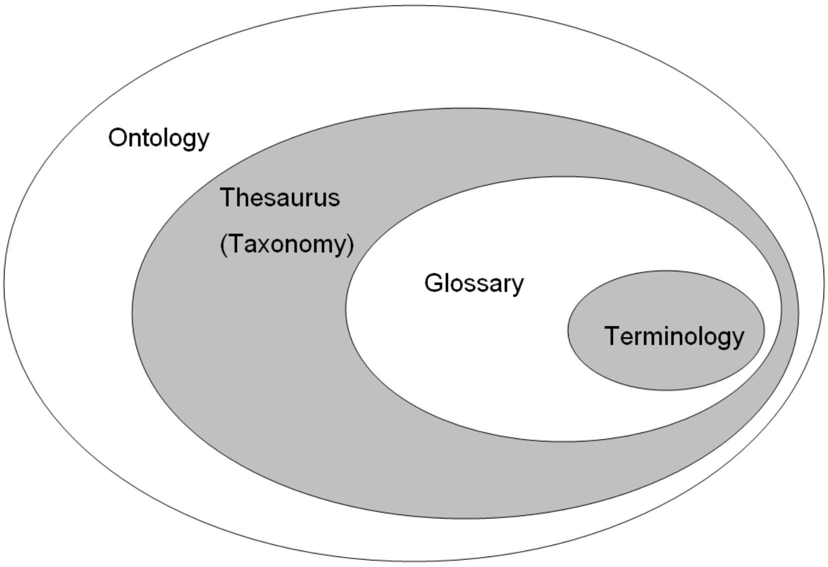


Figure 1.1: Knowledge Organization Systems

1.2.1 Controlled vocabularies and glossaries

In Library and information science controlled vocabulary is a carefully selected list of words (Terminology) and phrases, which are used to fill units of information so that they may be more easily retrieved by a search. Controlled vocabularies solve the problems of homographs, synonyms and polysemes by ensuring that each concept is described using only one authorized term and each authorized term in the controlled vocabulary describes only one concept. In short, controlled vocabularies reduces ambiguity inherent in normal human languages where the same concept can be given different names and ensures consistency.

A glossary is a list of terms in a particular domain of knowledge with the definitions for those terms. Traditionally, a glossary appears at the end a book and includes terms within that book which are either newly introduced or at least uncommon. A *core glossary* is a simple glossary or defining dictionary which enables definition of other concepts, especially for newcomers to a language or field of study. It contains a small working vocabulary and definitions for important or frequently encountered concepts, usually including idioms or metaphors useful in a culture. In computer science, a core glossary is a prerequisite to a core ontology. An example of this is seen in the Suggested Upper Merged Ontology or SUMO[1], an upper ontology (which describes very general concepts that are the same across all domains) intended as a foundation ontology for a variety of computer information processing systems.

1.2.2 Thesauri and Taxonomies

The term taxonomy has been widely used and abused to the point that when something is referred to as a taxonomy it can be just about anything, though usually it will mean some sort of abstract structure. Taxonomies have their beginning with Carl von Linné, who developed a hierarchical classification system for life forms in the 18th century which is the basis for the modern zoological and

[1] http://suo.ieee.org/SUO/SUMO/

botanical classification and naming system for species. A possible dfinition of taxonomy could be a monohierarchical classification of concepts. In reference to Web sites and portals, a site's taxonomy is the way it organizes its data into categories and subcategories, sometimes displayed in a site map.

Thesaurus is a controlled vocabulary in which concepts are represented by preferred terms, formally organized so that paradigmatic relationships between the concepts are made explicit, and the preferred terms are accompanied by lead-in entries for synonyms or quasi-synonyms). It is a taxonomy with extras relations, can be polyhierarchical, and contains scope notes to indicate exactly what the term means. While a taxonomy is designed to classify things, a thesaurus is designed to help you find the right words or phrases to describe what you are ultimately looking for. A thesaurus, on the other hand, emphasizes other aspects. It is basically a network of interrelated terms within a particular domain, and although it will often contain other information (such as definitions, examples

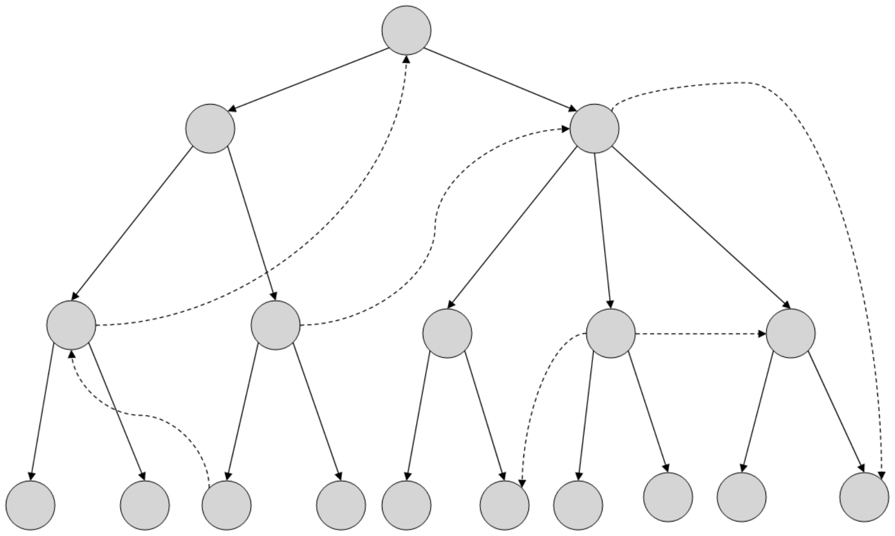


Figure 1.2: Taxonomy vs Thesaurus

of usage, etc.), the key feature of a thesaurus is the relationships, or associations, between terms (see Figure 1.2). Given a particular term, a thesaurus will indicate which other terms mean the same, which terms denote a broader category of the same kind of thing, which denote a narrower category, and which are related in some other way. An example of thesaurus resource is GEMET [2], the GEneral Multilingual Environmental Thesaurus, that has been developed as an indexing, retrieval and control tool for the European Topic Centre on Catalogue of Data Sources and the European Environment Agency. The basic idea for the development of GEMET was to use the best of the presently available excellent multilingual thesauri, in order to save time, energy and funds. GEMET was conceived as a "general" thesaurus, aimed to define a common general language, a core of general terminology for the environment. Specific thesauri and descriptor systems (e.g. on Nature Conservation, on Wastes, on Energy, etc.) have been excluded from the first step of development of the thesaurus and have been taken into account only for their structure and upper level terminology.

1.2.3 Semantic Networks

Glossaries, taxonomies and thesauri are all ways of mapping the knowledge structures that exist implicitly. In the field of AI (Artificial Intelligence) there also exists the need to be able to represent knowledge (and meaning), in order to support communication between people and machines. One widely used knowledge representation formalism is that of conceptual graphs, whose building blocks are concepts and conceptual relations.

Semantic networks are knowledge representation schemes involving nodes and links (arcs or arrows) between nodes. The nodes represent objects or concepts and the links represent relations between nodes. The links are directed and labeled; thus, a semantic network is a directed graph. Graphically, the nodes are usually represented by circles and the links are drawn as arrows between the circles as in Figure 1.3. This represents the simplest form of a se-

[2] http://www.eionet.europa.eu/gemet

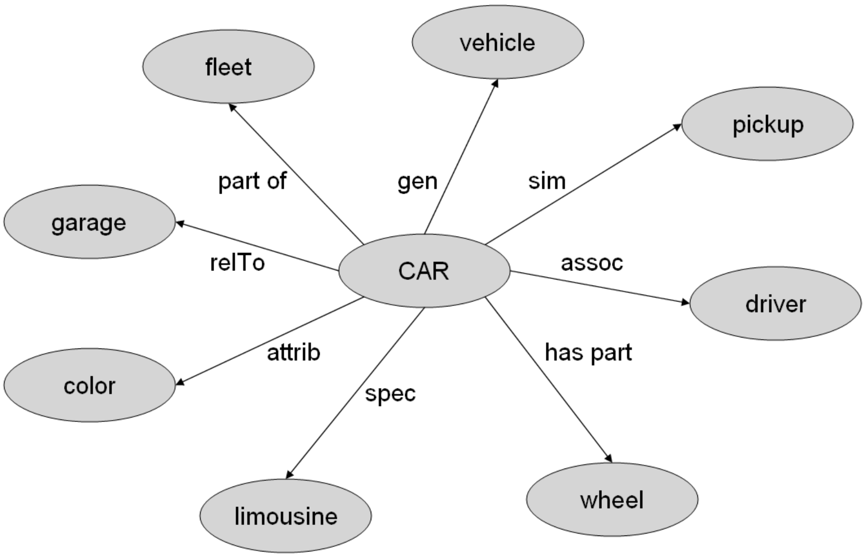


Figure 1.3: Semantic Network

mantic network, a collection of undifferentiated objects and arrows. The structure of the network defines its meaning. The meanings are merely which node has a pointer to which other node. The network defines a set of binary relations on a set of nodes.

1.2.4 Ontologies

The term 'Ontology' has been coined in philosophy to refer to basic existential issues and then has been adopted also by the artificial intelligence (AI) and the knowledge management research. A general but sound and complete definition of ontology is the following: "a formal, explicit specification of a shared conceptualization" [Gru93]. It is a conceptualization; indeed it represents a conceptual model of a specific domain. It is formal because it must be machine-understandable and processable, explicit because it needs to be defined in an unambiguous way and shared because it must be commonly accepted by a community of users that refer to it. Ontologies are typically composed by three kinds of entities [JDW06]:

1. a **set of concepts** that characterize the considered domain;
2. a **set of relations** between those concepts;
3. a **set of instances** of particular entities along with their specific properties.

Ontologies could be expressed adopting different formalisms or description languages: a formalism is a collection of various constructs useful to support the formal description of a particular domain of interest. When we choose a formalism we have to determine the right trade-off between two main opposite needs: its **expressive** power and its **complexity of reasoning** [Fra05].

The expressive power is the richness of different available constructs that could be exploited to describe a particular domain of interest: for instance we could mention the possibility to precisely define the properties of every concept or relation or to express more or less complex constraints. Closely related to the expressive power issue, other important properties that should be considered speaking about a formalism, concerning the possibility to entail new knowledge from that already stated, are:

- the **correctness** of entailment procedure: the impossibility to draw false entailed conclusions;
- the **completeness** of entailment procedure: the ability to draw all correct conclusions;
- the **decidability** of entailment problem: the existence of an algorithm which compute the entailed knowledge in a finite number of steps.

The complexity of reasoning is the more or less great amount of computing resources needed to obtain new entailed knowledge through a specific reasoning algorithm which applies a particular set of reasoning rules or procedures.

Usually if we increase the expressive power of a formalism, it will better describe a particular domain, but also its complexity of reasoning will parallelly grow and thus its ease and directness of use will decrease, making more difficult to take advantages of it. Moreover, increasing the expressive power of a formalism, it could happen that it looses its decidability or its completeness and correctness.

1.3 The Semantic Web

Nowadays when we consider the Web we think of a great deal of linked data regarding a huge number of different knowledge domains; all this available data has a great informative value, but many times it is not possible to take real advantages of it because of its different and often chaotic organization and structure. There is a lack of ease of integration and interoperability between multiple different sources of information and usually these tasks are very expensive to realize and tailored to specific situations.

The Semantic Web represents an attempt to gradually solve this kind of problems adding richer and more structured descriptive content to Web data. The greatest part of actual Web content is built exclusively for humans, meaning that most of Web information is available as textual content, opportunely structured in order to be rendered by Web browsers, usually preventing users form deciding the way it is presented and used. Sometimes it is also possible to obtain different and more synthetic representation of Web data, but we need to adapt our applications to manage these particular representations in order to really exploit it. The Semantic Web aims at going beyond the textual level of content delivery. As stated by Tim Berners-Lee , its major deviser in 2001 [BLTO01], the Semantic Web 'is an extension of the current Web in which information is given a well defined meaning, better enabling computers and people to work in cooperation'. It represents an attempt to integrate actual Web with all the possibilities offered by the semantic characterization and elaboration of information, deeply involving the semantic knowledge management and artificial intelligence (AI) fields [SWH06]. It aims at the production of additional metadata in order to express Web informative content in a universal and machine-understandable way: as a consequence every source of information over the Web should be able to express the meaning of its data in order to make them highly and easily reusable and integrable in different contexts. In fact one of the major general defining characteristics of the Semantic Web is the serendipity [KeySem2006]: it represents the possibility to integrate and reuse the information contained in different heterogeneous systems, devices and services without knowing anything about at design time but only exploiting the support provided by the semantic description of the exposed data. After having introduced a strong and diffused semantic global infrastructure over the Web, it should be possible to use software agents in order to interpret, aggregate and filter Web data considering user goal, preferences and context and exploiting some automated reasoning techniques and rules, already extensively used in AI.

In order to semantically describe Web informative content and process it, the Semantic Web relies on two fundamental theoretical bases: the ontologies and, in particular, the description logic formalism and its reasoning possibilities. They represent the formal ground used to describe knowledge in an unambiguous way so as to expose it and to make possible some sort of inference of new knowledge from that already stated. As a consequence many new issues have arisen, mainly regarding the management of multiple ontologies in a highly distributed and heterogeneous context like that one represented by the Web.

Starting also from this formal foundation, during the last few years the World Wide Web Consortium (W3C) [w3c] has developed and standardized the Resource Description Framework (RDF) [Rdf] which represents the Semantic Web fundamental knowledge representation language: it allows expressing machine-processable statements about URI-referencable resources, using a triple-based model. The widely diffused XML syntactic meta-language is used to serialize and exchange RDF data. In parallel with RDF, in order to make possible to exploit its high expressive power and versatility, another related language has been defined by the W3C: the Resource Description Framework Schema. It provides many facilities to define RDF vocabularies (basic ontologies) in terms of specific sets of classes, properties and some simple usage constraint; they are used as a descriptive reference to represent a particular domain of knowledge in order to express factual RDF statements.

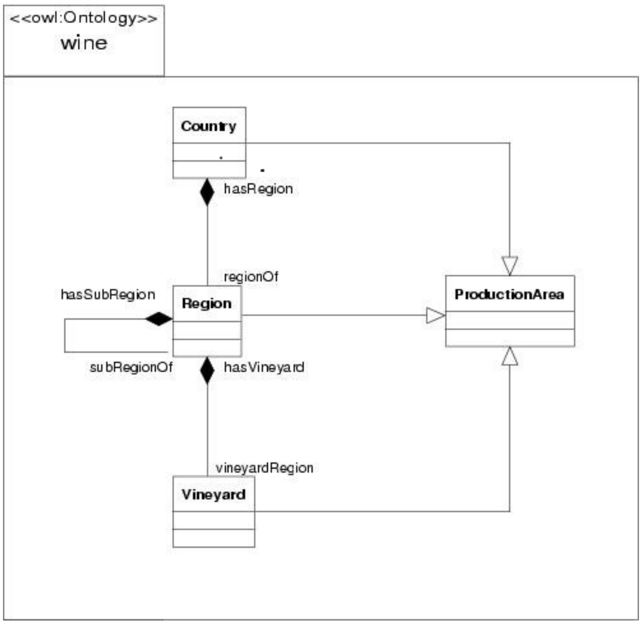


Figure 1.4: Representation of OWL ontology

After RDF and RDFS introduction, in reply to the need for major expressive capabilities when describing the structure of knowledge domains, the W3C has developed and standardized a new description language: the Web Ontology Language (OWL)[owl]. It is an extension to RDFS. It adds many other expressive possibilities to better define the conceptual structure of a particular domain or, in other words, to strongly structure a domain ontology (see Figure 1.4). Summarizing, RDFS and OWL both represent means to define reference model in order to express factual RDF assertions.

As a consequence of the adoption of RDF as the universal way to make statements about resources over the Web, the need to define a new language to query RDF data collections has arisen. The W3C has developed SPARQL Query Language for RDF [wsp]. It is a first attempt towards the standardization an RDF query language; at the moment it is still a W3C Working Draft.

Nowadays many research efforts related to the Semantic Web are affecting different application areas: the semantic annotation of resources, the semantic support to information browsing and search, the improvements of actual Web Services infrastructure by adding semantic capabilities and so on. Also a growing number of tools and frameworks to semantically structure, manage and store information is being developed.

At present Semantic Web technologies are mainly diffused in the world of corporate intranets or in very specific communities of Web users. Some RDF reference vocabulary along with the related RDF data are also spreading over the global Web, being more and more adopted, but a killer application that really and globally introduces these new way of organizing and sharing information is still absent.

1.4 Computational Lexicons

Computational lexicons include manipulable computerized versions of ordinary dictionaries and thesauruses. Computerized versions designed for simple lookup by an end user are not included, since they cannot be used for computational purposes. The term 'Lexicons' can denote any electronic compilations of words, phrases, and concepts, such as word lists, terminology databases, glossaries, taxonomies, thesauri, wordnets, and ontologies. In general, a *lexicon* includes a wide array of information associated with entries. An *entry* in a lexicon is usually the base form of a word, the singular for a noun and the present tense for a verb. Using an ordinary dictionary as a reference point, an entry in a computational lexicon contains all the information found in the dictionary: inflectional and variant forms, pronunciation, parts of speech, definitions, grammatical properties, subject labels, usage examples, and etymology. More specialized lexicons contain additional types of information. A semantic lexicons such as Wordnet [Wne] contains synonyms, antonyms, or words bearing some other relationship to the entry. A bilingual dictionary contains translations for an entry into another language.

It is not easy to classify the lexical resources in more overall groups because each resource often contains different mixtures of data. For example, it seems obvious to group the Princeton WordNet with the EuroWordNet [Vos04] data as semantic networks but EuroWordNet also contains equivalence relations, making it a multilingual resource, and it incorporates a formalized ontology, making it partly a conceptual knowledge base.

The term *computational* applies in several senses for computational lexicons. Generally, a lexicon is any dictionary in an electronic form. Firstly, the lexicon and its associated information may be studied to discover patterns, usually for enriching entries. Secondly, the lexicon can be used computationally in a wide variety of applications; frequently, a lexicon may be constructed to support a specialized computational linguistic theory or grammar.

1.4.1 Dictionaries

The simplest form of a computational lexicon is a dictionary. It lists all lexical entities in a domain, connects them with their semantic meaning via a defining gloss, and enumerates all senses in case of polysemous entities. Words can appear in many different forms, but only the *lemma* form appears as the main word or headword in most dictionaries. Many dictionaries also provide pronunciation information, grammatical information, word derivations, histories, etymologies, illustrations, usage guidance and examples in phrases or sentences. Most dictionaries are produced by lexicographers and are most commonly found in the form of a book. Today, dictionaries are also found in electronic form and there are around one thousand english on-line dictionaries available on the web.

There are also a lot of multi-source dictionaries on the Web whereof the most popular are:

- **OneLook.com** is a dictionary search service that provides a way to look up words and phrases at multiple sites. It is a search aggregation service for dictionaries, glossaries, and encyclopedias functioning as a form of search engine. It allow to search references for words that have definitions conceptually similar to the words using a statistical processing. More than 7 million words in more than 900 online dictionaries are indexed by the OneLook search engine.

- **FreeDictionary.com** is one of the best dictionary resource on the World Wide Web. We use multiple sources of data and a large part of this information is checked and edited by humans staff. This dictionary was derived from the Webster's Revised Unabridged Dictionary [3] and from WordNet and is being updated and supplemented by an open coalition of volunteer collaborators from around the world. This electronic dictionary is the starting point for an ongoing project to develop a modern on-line comprehensive encyclopedic dictionary, by the efforts of all individuals willing to help build a large and freely available knowledge base. Contributions of data, time, and effort are requested from any person willing to assist creation of a comprehensive and organized knowledge base for free access on the internet.

Another very interesting project is Wiktionary, a collaborative project to produce a free, multilingual dictionary with definitions, etymologies, pronunciations, sample quotations, synonyms, antonyms and translations. Wiktionary is the lexical companion to the open-content encyclopedia Wikipedia and his English edition currently have 119,091 entries. Unlike standard dictionaries, it is written collaboratively by volunteers using wiki software, allowing articles to be changed by almost anyone with access to the Web site. Because Wiktionary is not limited by print space considerations, most of Wiktionary's language editions provide definitions

[3] http://machaut.uchicago.edu/websters

and translations of words from many languages, and some editions offer additional information typically found in thesauruses and lexicons.

Dictionary	Words
Wordnet 3.0	155,287
Infoplease Dictionary	127,110
Wiktionary	119,091
Encarta	107,301

Table 1.1: Examples of on line dictionaries

1.5 WordNet

Currently the most relevant Web-accessible lexical resource is WordNet [Wne]. It is a lexical reference system that explicitly represents many different characteristics of the human linguistic knowledge. It has been conceived in 1985 by a group of research of the Princeton University, on the basis of psycholinguistic theories concerning human memory. Since then its contents and its terms' coverage and relations have been continuously enriched; also the structure of the language representation model has been improved and better defined. WordNet is updated by a group of lexicon experts [GAMM93].

In this section we describe the fundamental features related to WordNet language representation and accessibility. Then we better analyse the growing attention that WordNet has received by the Semantic Web community, mentioning the possible supports that this lexical resource can provide to the Semantic Web activities. In conclusion we briefly describe an important issue that must be considered dealing with open dynamic environments like the Web: the support and interoperability between multiple languages. In particular, we consider the Inter-Lingual-Index mechanism.

1.5.1 Structure and usage

WordNet language structuring is based on the fundamental distinction between [GAMM93]:

- **lexical form** : it is the way used to represent a single word as a sequence of characters (string);

- **meaning** : it is a specific concept; it can be referred using one or more different word forms.

The many-to-many relations between meanings and lexical forms could be represented by a lexical matrix. It is a sort of table in which every row corresponds to a particular meaning and every column to a specific lexical form. Every lexical form can represent different meanings; in this case it is a polysemous one. For example the lexical form 'car' can represent two different meanings: a four wheel vehicle and the machine where passengers ride up and down. On the other end, every meaning can be expressed through different lexical forms that are called synonyms. For instance, the lexical forms 'machine' and 'car' can both refer to a four wheel vehicle. In Figure 1.5 an examples of **lexical matrix** is represented, underlying the occurrence of synonymy and polysemy.

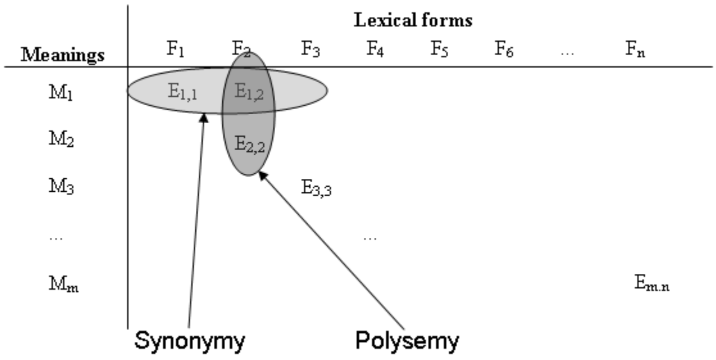


Figure 1.5: WordNet lexical matrix (meanings and lexical forms).

Starting from the lexical matrix representation we can expose the basilar entity that constitutes the core of WordNet: the synset. A synset represents a specific meaning or concept and is identified by the set of synonym lexical forms that can be used to refer to that particular meaning. For example the lexical forms car, auto, automobile, machine and motorcar constitutes the synset that define the concept of four wheels vehicle.

Starting from the meanings and the lexical forms we can define the WordSense: it is the association of a lexical form to a particular meaning identified by a synset, thus determining one of the different concepts referable using that lexical form. Every element of the lexical matrix in Figure 1.5 represent a WordSense.

Starting from the elements of WordNet already described, we can list the following general organization considerations:

- the number of **WordSenses** generated by a synset is equal to the number of **lexical forms** it contains;
- every **WordSense** is associated exactly to a single **synset**;
- every **WordSense** is referred to a single **lexical form**;
- every **lexical form** can belong to one or more **WordSenses** and thus can be associated to one or more **synsets**.

In Figure 1.6 we graphically represent the cardinality of the relation between Synsets, WordSenses and lexical forms.

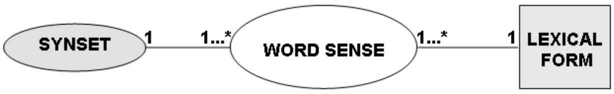


Figure 1.6: Cardinality of the relations between synsets, WordSenses and lexical forms.

In Figure 1.7 we show an example of different synsets sharing lexical forms; each synset is identified by the description of its meaning, called gloss. The abbreviation WS stands for WordSense;

it is the result of the intersection between a lexical form (rectangle) and a meaning (circle).

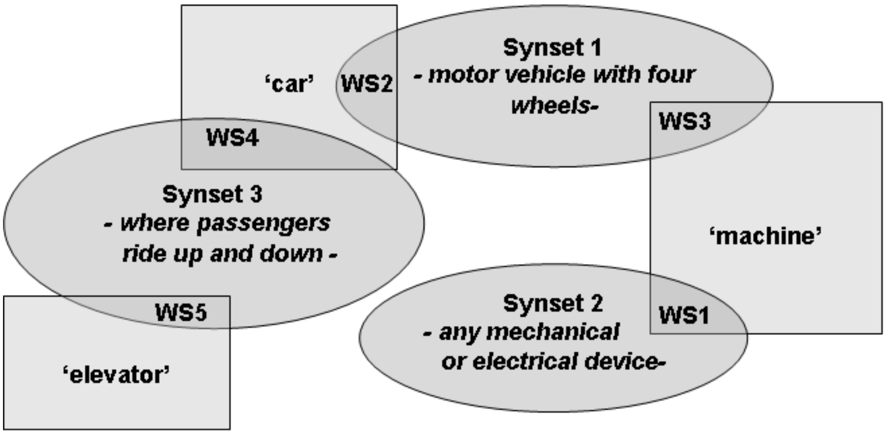

'car'
WS2
Synset 1
- motor vehicle with four wheels-
WS3
WS4
Synset 3
- where passengers ride up and down -
'machine'
WS5
Synset 2
- any mechanical or electrical device-
WS1
'elevator'

Figure 1.7: Example of different synset sharing lexical forms.

In WordNet are considered four parts of speech: **nouns**, **verbs**, **adjectives** and **adverbs**. Every concept is associated to a particular part of speech (POS).

Two different general kinds of relations have been defined in order to represent the associations that characterize our mental representation of linguistic structures:

- **lexical relations**, between two or more WordSenses;

- **semantic relations**, between two synsets or meanings.

In what follows we provide some significant example of lexical and semantic relations. Among the lexical relations there are:

- **Synonymy** : it connects all the WordSenses that refer to the same meaning thus constituting a synset;

- **Antinomy** : it connects two WordSenses referring to opposite meanings, for instance 'natural_object' and 'artifact';

- **SeeAlso** : it links a WordSense with one or more other ones that can provide further descriptive information; for example the verb 'breathe' can be connected to the verbs 'breath_in' and 'breath_out';
- **Participle** : it associates an adjective WordSens (participle form) with the verb WordSense from which it derives; for example it links the adjective 'applied' with the verb 'apply'.

Some significant semantic relations are:

- **Hypernomy / hyponymy** : they respectively represent relatins of generalization / specialization between concepts or synsets; for example the concept of 'station_wagon' is a specialization or an hyponym of the concept of 'car' and, inversely, the concept of 'car' is a generalization or an hyperonym of the concept of 'station_wagon';
- **Meronymy / olonymy** : this relation is used to represent part-whole associations between concepts: a 'call' is a part or meronym of an 'organism' and, inversely, an 'organism' is an olonym of a 'cell' meaning that it is composed by cells;
- **Entailment** : it is a relation between two verb concepts: the former implies the latter if and only if the latter can be executed if the former is not. For instance the verb 'walk' entails the verb 'step';
- **Attribute** : it links a name concept with one or more adjective concept that express possible characteristics of that name: the name 'measure' can be connected with the adjectives 'standard' and 'non_standard';
- **Similarity** : it associates two adjective concept with similar meaning; for instance 'wet' with 'moist' or 'dry' with 'arid';
- **Same verb group** : this relation links two verb concepts with an analogous meaning; for example the verb 'breath' with 'respire'.

Every relation can be symmetric and/or transitive and can be characterized by restrictions regarding the parts of speech that it connects.

At present, WordNet version 3.0 (see Table 1.2) is available; it includes 120982 concepts (or distinct synsets).

POS	Words	Synsets	Word-Sense Pairs
Nouns	117798	82115	146312
Verbs	11529	13767	25047
Adjectives	21479	18156	30002
Adverb	4481	3621	5580
Total	155287	120982	206941

Table 1.2: Number of Entries and Senses in Wordnet

There are different available formats to export and query WordNet lexical data collection. This lexical resource can be queried as a standalone desktop application thanks to an appropriate graphical WordNet browser available along with database files. Moreover, WordNet data collection is available as a Prolog database (referred to as Prolog distribution) opportunely structured and organized in different files. WordNet queries can be also executed directly over the Web exploiting an HTML form-based interface.

The traditional applications of WordNet lexical information are mainly related to different kinds of automated text analysis [Lit97] [JMM04]. Word sense disambiguation, the process of automatically deciding which sense is intended for a particular word in a given context, can be supported by the huge amount of lexical relations included in WordNet. Also information extraction procedures, related to the automatic identification of selected types of entities, relations, or events in free text, can benefit from the huge amount of interconnected data that WordNet provides. In automated question answering, WordNet lexical contents are usually exploited to interpret the meaning of a user defined question and determine what type of answer is required. Many processes of automatic characterization and indexing of textual contents like documents, but also

Web pages use WordNet relations to evaluate the similarity between the contents of different text so as to cluster them so as to simplify their retrieval procedures.

Generalizing, all the applications just listed use the semantic lexical information contained in WordNet to support different tasks that need to define or process the meaning of textual contents in order to be executed. As a consequence the usefulness of the lexical resource considered is measured in terms of its completeness and its richness of useful semantic relations.

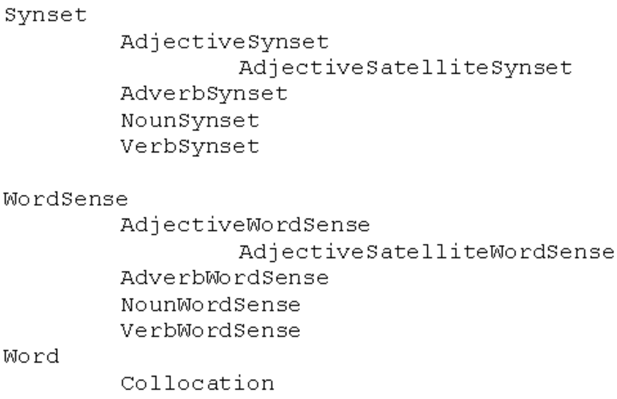
1.5.2 WordNet and the Semantic Web: RDF/OWL representation

During the last years, the lexical reference WordNet has received a growing attention by the Semantic Web research community. After the born in 2004 of a ‘WordNet Task Force’ of the W3C’s ‘Semantic Web Best Practices Working Group’ (SWBPWG) [bpw], WordNet has been translated in the widely adopted standard semantic languages RDF and OWL, and then has been published a Working Draft [wnw] as a rielaboration and a synthesis of existing non-standard conversions.

RDF Schema and OWL, designed to describe collections of resources on the Web, are convenient data models to represent highly interconnected information and their semantic relations, and therefore useful to support WordNet graph data model, composed by many synsets or WordSenses interconnected by different kinds of relations. Moreover RDF/OWL representation of WordNet is easy extensible, allowing for interoperability and making no assumptions about a particular application domain.

The conversion is based on the definition of an ontological description of the semantic structures that constitute WordNet lexical data; it consists in a hierarchy of classes and properties organized on the basis of the conceptual structure of the Princeton’s WordNet Prolog distribution. The reference’s conceptual model has been changed only in the representation format, without affecting the original architecture.

WordNet model is composed by three main classes: Synset, WordSense and Word. The first two are divided into four fundamental subsets, each of one related to a specific part of speech: noun, verb, adjective and adverb. The only subset of Word is Collocation, used to represent words that have hyphens or underscores in them; this classes' hierarchy is shown in Figure 1.8.



```
Synset
        AdjectiveSynset
                AdjectiveSatelliteSynset
        AdverbSynset
        NounSynset
        VerbSynset

WordSense
        AdjectiveWordSense
                AdjectiveSatelliteWordSense
        AdverbWordSense
        NounWordSense
        VerbWordSense
Word
        Collocation
```

Figure 1.8: WordNet RDF/OWL classes' hierarchy.

The properties:

- represent lexical relations between the main classes, connecting couples of Synsets or WordSenses;
- describe attributes of classes;
- connect each Synset with WordSense/s (*wn:synsetContainWordSense*) and each WordSense with the Word it represents (*wn:Word*).

Each Word is connected to its lexical form through the property *wn:lexicalFrom* and each Synset is characterized by a specific type (*rdf:type*) related to the part of speech considered; this scenario is represented in Figure 1.9.

This representation of WordNet, composed of a single RDF/OWL schema, provides OWL semantics while still being interpretable by

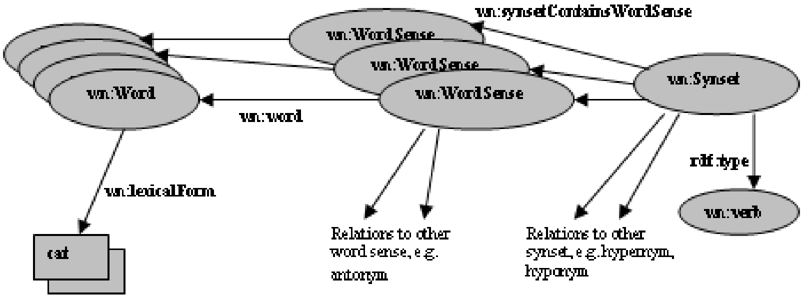

wn:synsetContainsWordSense
wn:WordSense
wn:WordSense
wn:WordSense
wn:Synset
wn:Word
wn:word
rdf:type
wn:lexicalForm
wn:verb
cat
Relations to other word sense, e.g. antonym
Relations to other synset, e.g. hypernym, hyponym

Figure 1.9: Diagram of the schema of Wordnet RDF/OWL (prefixes 'wn' and 'rdf' stand for respectively the namespaces of WordNet and RDF.

pure RDF Schema tools. Moreover, it defines a robust, human-readable URI assignment system in order to unambiguously reference every single entity contained in WordNet. An on-line querying model based on the Common Bounded Description of resources and a reduced version of WordNet database (called WordNet Basic), so as to keep the footprint small when the complete set of relations is not needed are also provided.

Along with the RDF/OWL ontology that describes the classes and properties which WordNet structure is based on, all the real lexical data are contained in 19 RDF files; each of them is used to define some particular kind of relation or data in order to provide an highly modular contents' collection.

The adoption of WordNet Web Services to support the RDF/OWL representation can represent another important step towards a stronger integration and an effective use of this important lexical resource into Semantic Web.

This kind of resources could have a fundamental place in many Semantic Web basic processes. WordNet can be used as general domain reference ontology to simplify the mapping of different specific ones. Also the semantic interpretation of Web Services [LSG05] can be supported exploiting the semantic content of WordNet. The annotation of every kind of resources is considerably improved if the added information refers to WordNet concepts; indeed the huge set of available semantic relations can be exploited to improve search possibilities and contents' organization; in [Bid03] is provided a possible WordNet usage: organization and annotation of photographs. Moreover, in the future the addition of interlingual information handling possibilities to RDF/OWL data model can support the achievement of real multiligual semantic interoperability in the Web. This topic has been explicitly left unsolved by W3C WordNet Task Force.

1.5.3 Multilingual support

WordNet structure can be used to define lexical resources each of them referring to a particular language. Currently, there are different WordNets concerning distinct languages: the Global WordNet association [gwn], constituted to foster interoperability between WordNets of distinct languages, comprehends 46 different WordNets in 38 different languages.

At present, the Web represents a global and highly distributed context in which many languages from all over the world must be managed in a unitary environment. In such a scenario the support for interoperability between different languages is fundamental. Moreover, the multilingual support is currently focusing increasing relevance, along with the growing diffusion and integration of lexical resources, in particular of different WordNets over the Web so as to support Semantic Web vision.

In order to introduce some support for the interoperability between WordNets of different languages, in 1996 was born the EuroWordNet project [ewn]. Currently it is concluded; it has produced a sort of collection of mapping data between WordNets of different languages. Each mapping is obtained through the InterLingual Index (ILI). It is a collection of many entries; each of them include a synset, a short definition of the concept intended and the reference to the corresponding resource in the English version of WordNet.

In Figure 1.10 a schematization of the ILI structure is provided; the ILI is composed by four records that are used to map the synsets (represented by circles) of three WordNets expressed using different languages. The ILI mapping relations are visualized by the dashed arrows; the other arrows point out common internal WordNet relations.

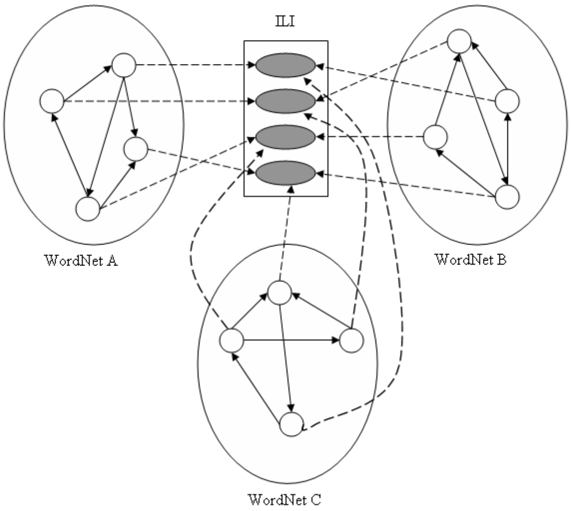


Figure 1.10: WordNet InterLingual Index (ILI) structure.

Four kinds of relations are used to map a synset of a particular WordNet to a specific ILI record:

- **EQ_SYNONYM** : complete equivalence of the two concepts, one belonging to the WordNet considered and the other related to a particular ILI record;
- **EQ_NEAR_SYNONYM** : the concept of the WordNet considered is mapped to more than one concept identified by an ILI record or vice versa;
- **EQ_HAS_HYPERONYM** : the concept of the WordNet considered is more specific than every other concept identified by an ILI record available;

- **EQ_HAS_HYPONYM** : the concept of the WordNet considered can be related only to ILI records referring to more specific concepts.

Also some simple kind of relation between ILIs is provided in order to maximize the Multilanguage mapping possibilities.

The ILI mapping provides the possibility to independently develop every single WordNet, making it interoperable with others simply mapping its synsets on the corresponding ILI record.

Currently, the Euro WordNet project is continued through the Global WordNet association [gwn]; it was founded in 2000 and represents the most relevant intiative taht aims at supporting multilingual interoperability between WordNets; it is a non-profit organization that wants collect the different WordNets referred to different languages in order to unify them and allow for the development of methodologies, standard procedures and shared representations to support their interactions.

1.6 The Social Web

The Social Web is a broad-meaning term that concerns every kind of interaction and every form of collaboration between Web users. The Web has always represented a mean to create direct and indirect social connections between people. Along with the increasing number of Web users and the evolution of Web technologies, during the last few years, many new user interaction and collaboration patterns have been introduced supporting improved paradigms of communication. Besides the classical methods of interaction, like e-mail, chat, newsletter discussion boards, new ones are experimenting growing diffusion: social networking and virtual communities, blogs, wikis, collaborative editing and tagging systems represent only some example [ssw] [Wro06].

We focus on the **collaborative content construction** that is related to the collaboration of many users to gather shared information in order to provide useful services to an entire community.

Many social Web services include both kinds of users' interactions or collaboration patterns.

In what follows we analyse the social aspects of the most relevant currently collaborative systems:

- **Wikis** are *collaboratively authoring of documents*; this kind of new tools allows for collaborative content construction. Every user can give his contribution to the growing of the information contained in the Wiki in order to make it more useful to the entire community. Wikipedia [Wika], one of the most visited and relevant current Web resources, is a free multilingual encyclopaedic collection of collaboratively edited information; it represents the outcome of a worldwide editing effort of a huge amount of users/authors that constantly update its contents;

- **Social tagging systems** are services in which every registered user can tag some sort of Web resource associating one or more freely chosen keyword. Nowadays there are many different tagging services concerning keyword-based description of URL-referencable contents (social bookmarking), videos, photos, blog posts, etc. The three main components of collaborative tagging systems are: users, resources and tags [LA05]. Users may be connected in groups with common interests; resources may be related by the different kinds of links which constitute the basis of current Web; tags provide the connection between a single user and a particular resource. Tagging services mainly are devoted to the *collaborative content construction* thanks to their collection of a growing amount of descriptive information about resources that allows for better management and improved searches' effectiveness; many times communities of users are created around a particular tagging tool;

- **Contents sharing tools** allows users sharing some sort of data: photos, videos and so on. Every user can usually access to the global collection of *shared contents* and also contact the author, thus creating *user-to-user interactions*. A famous example of content sharing service is represented by YouTube [you]. It is owned by Google and allows freely sharing videos; users can upload, view, and share video clips. Videos can be rated; the average rating and the number of times a video has been watched are both published. At present it is one of the most popular Web sites with an avarage of 100 million clips daily viewed an 65.000 daily new uploads;

- **Collaborative document editing tools** allow editing a text by different participants over the Web, managing concurrent accesses and the changes' updates of multiple versions. Nowadays they usually offer also advanced layout definition possibilities. They allows for *collaborative content construction*, even if among a usually small group of users/editors. One of the most popular collaborative document editing tool is Google Docs & Spreadsheets [gde]: a Web-based word processor and spreadsheet application that make it possible to create and edit documents and spreadsheets online while collaborating in real-time with other users.

When we speak about the **collaborative creation of shared contents** we must consider the different motivations that push users to this activity: their ego, their reputation, their dreams of fame and riches are important factors, but also their passion for the subject of matter and their recognition as members of a community must be pointed out [You06]. The awareness that their efforts represent a contribution to the creation of a huge collection of data that can be of great usefulness to each one of them represents also a fundamental motivation. As a consequence the enrichment of a shared collection of free contents is seen as a sort of **social effort**. This kind of considerations is very relevant dealing with Wikis and social tagging systems. In particular, in **collaborative tagging services**, when every user can assign a freely defined set of tags to a resource, the tag collection will reflect the social attitudes of the community of users and a shared social organization and structuring of the tag-space will emerge: this phenomenon is referred to as **emergent semantics**. It continuously adapts the tag space to the way users choose to describe resources, reflecting their tagging behaviour. The result of this process of adaptive social structuring of the tag-space in a collaborative tagging system has recently been defined as **folksonomy** [Smi04] .

Part II

Management of Semantic Resources

Chapter 2

An Architecture for developing Semantic Resources

Abstract

In this chapter we want to present a general architecture for the management of semantic resources. For 'management' we intend not only create and edit resources but also integrate existing ones by using approaches of cross-fertilization. Today we can observe a proliferation of multilingual, Web-based, semantic resources. Moreover, market calls for new types of semantic resources, rapidly built and easy tailored, exploiting the richness of existing resources. To meet these needs, semantic resources need to be made available, to be constantly accessed by different types of users, who may want to select different portions of the same resource, or may need to combine information coming from different resources. In this chapter we present a three-layer architecture able to turn into reality the vision of shared and distributed semantic repositories.

2.1 Sharing semantic resources

The emerging social and cultural phenomena of multilingual, Web-based, machine-readable, free content semantic resources no longer leaves space to static, closed, and locally managed repositories of semantic information. Instead, it calls for an environment where semantic resources can be shared are reusable, and are openly customizable. At the same time, as the history of the web teaches, it would be a mistake to create a central repository containing all the shared resources because of the difficulties to manage it. Distribution of resources thus becomes a central concept: the idea consists in moving towards distributed general-purpose resources, based on open content interoperability standards, and made accessible to users via web-services technologies. There is another, deeper argument in favor of distributed semantic resources: language resources, lexicons included, are inherently distributed because of the diversity of languages distributed over the world, that makes it impossible to have one single centralized repository of resources. In this way, each resource is developed and maintained in its natural environment. This new type of semantic resources can still be stored locally, but its maintenance and exploitation can be a matter of agents being choreographed to act over them.

2.2 Orchestration

A possible solution of this problem can come from collaborative tools such as application for workflow orchestration. Admittedly, this is a long-term scenario requiring the contribution of many different actors and initiatives (among which we only mention standardization, distribution and international cooperation). The first prerequisite for this scenario to take place is to ensure true interoperability among semantic resources, a goal that is long being addressed to by the standardization community and that is now mature. Although the paradigm of distributed and interoperable lexical resources has largely been discussed and invoked, very little has been made for the development of new methods and techniques for its practical realization. Some initial steps are made to design frameworks enabling interlexica access, search, integration and operability. An example is the Lexus tool [MKSW06], based on the Lexical Markup Framework [LRSA06], that goes in the direction of managing the exchange of data among large-scale lexical resources. A similar tool, but more tailored to the collaborative creation of lexicons for endangered language, is SHAWEL [GH02]. However, the general impression is that little has been made towards the development of new methods and techniques for attaining a concrete interoperability among semantic resources.

2.3 Three-layer architecture

In order to overcome limits of singular resources and to reach a common knowledge platform, a change in the very basic assumptions on the design, creation, maintenance and distribution of knowledge resources is needed and in this sense very alluring suggestions come from the web. In particular, the emerging operational and theoretical paradigm based on the notions of cooperation, collaboration and social knowledge determination seems to offer a way to rethink the entire strategy of creation of semantic entries. Collaboration is what subtends the practice of groups asynchronously producing works together through individual contributions (in the so-called collaborative authoring).

Designing a general architecture able to turn into reality the vision of shared and distributed semantic repositories is a very challenging task. We designed a distributed architecture to enable a rapid prototyping of cooperative applications for developing semantic resources. This architecture is articulated in three layers:

1. A higher layer, called cooperative layer or LeXFlow (see chapter 4) is intended as an overall environment where all the modules realized in the lower layers can be integrated in a comprehensive workflow of human and software agents. This layer was built on XFlow, a framework cooperative management of XML resources that has been developed during the

first part of my research activity in the Institute of Informatics and Telematics (IIT) of the National Research Council (CNR) of Pisa. (see Chapter 3) Borrowing from techniques used in the domain of document workflows, we model the activity of lexicon management as a particular case of workflow instance, where lexical entries move across agents and become dynamically updated. To this end, we have designed a lexical flow (LF) corresponding to the scenario where an entry of a lexicon A becomes enriched via basically two steps. First, by virtue of being mapped onto a corresponding entry belonging to a lexicon B, the entry(LA) inherits the semantic relations available in B. Second, by resorting to an automatic application that acquires information about semantic relations from corpora, the relations acquired are integrated into the entry and proposed to the human encoder. As a result of the lexical flow, in addition, for each starting lexical entry(LA) mapped onto a corresponding entry(LB) the flow produces a new entry representing the merging of the original two.

2. The middle layer hosts some applications that exploit the semantic shared repositories. The so-called "multilingual WN Service" Service (MWS) allows to mutually enrich wordnets in a distributed environment (see chapter 5). This module is responsible for the automatic cross-lingual fertilization of lexicons having a WordNet-like structure. Put it very simply, the idea behind this module is that a monolingual wordnet can be enriched by accessing the semantic information encoded in corresponding entries of other monolingual wordnets. Since each entry in the monolingual lexicons is linked to the Interlingual Index (ILI), a synset is indirectly linked to another synset in another and on the basis of this correspondence, can be enriched by importing the relations. The SemKey prototype is another application that exploit the shared ontology for disambiguating keywords (see Chapter 6). Other, more advanced NLP applications (in particular multilingual) can be developed by exploiting the availability of the repositories.

3. The lower layer consists of a sort of a grid of local services realized as a virtual repository of XML databases residing at different locations and accessible through web services. Basic software services are also necessary, such as an UDDI server for the registration of the local wordnets and web services dedicated to the coherent management of the different versions of WordNet the databases refer to.

The Figure 2.1 illustrates the general architecture. In this thesis we concentrate on the description of cooperative layer and the middle layer.

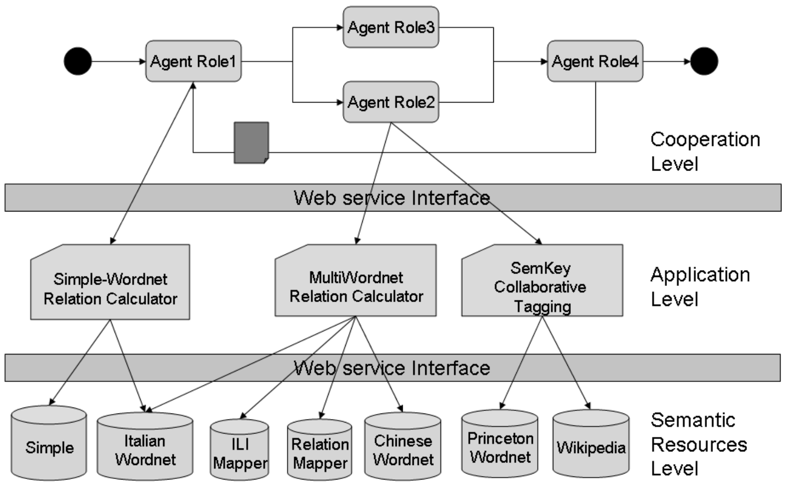


Figure 2.1: Three-Layer Architecture

Chapter 3

XFlow: the core of LexFlow

Abstract

Workflow management [GHS95] [SJHB96] involves the modeling and enactment of workflows. A workflow is either a basic workstep (called activity) or a complex workflow that consists of further workflows. In this chapter we describe XFlow is a framework for cooperative management of documents where the cooperation is driven by data contained in the documents and a set of procedural rules. The problem of processing documents has long been recognized to be a critical aspect in the enterprise productivity ([GHS95], [KRSRR97], [BCC$^+$99], [KMK02]). The management of documents becomes more difficult when it involves different actors, possibly in a decentralized working environment, with different tasks, roles and responsibilities in different document sections. XFlow is based on the paradigm of workflow management systems (WFMS) and for these reasons basic workflow management concepts are introduced and we present a workflow classification schemes that have been proposed in the literature. Workflows are described by means of a new XML language called XFlowML (XFlow Markup Language) largely based on XSLT Processing Model. XFlowML describes the document workflow using an agent-based approach.

3.1 Workflow Classification

A common way to classify workflows originates from the trade press, where ad hoc, collaborative, administrative, and production workflows are distinguished [GT98]. These four kinds of workflows are categorized according to their business value and their repetitiveness.

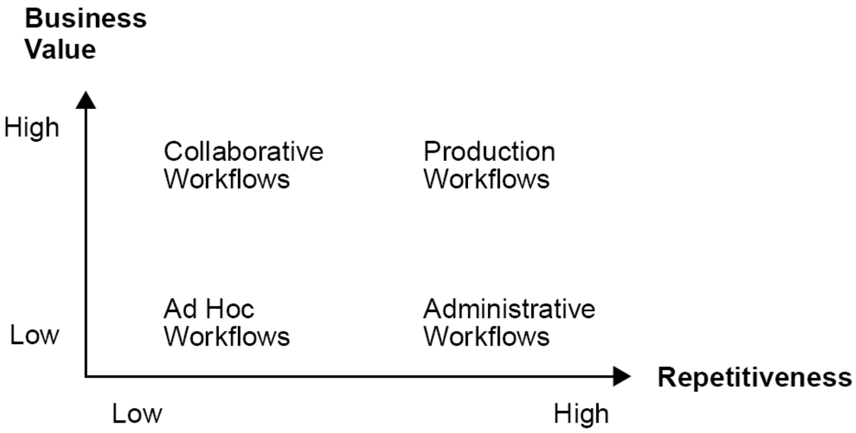


Figure 3.1: Workflow Classification

The business value of a workflow expresses the importance of the workflow to the organization within the workflow is carried out. Workflows with a high business value are typically concerned with the organizations core business. The repetitiveness of a workflow indicates how often the workflow is carried out in a similar way. Since creating workflow types is often an expensive task, only the development of workflow types for workflows with a high repetitiveness is economically justifiable. Using the criteria of business value and repetitiveness, four kinds of workflows can be categorized as shown in Figure 3.1. Ad hoc workflows and collaborative workflows typically involve humans collaborating in order to reach a certain goal. For these workflows, usually no predefined procedures and patterns exist, i.e., repetitiveness of these workflows is low, and therefore no workflow type can be defined in advance. The main difference between ad hoc workflows and collaborative workflows is their business value. Collaborative workflows have a high business value and include such tasks as the preparation of product documentation or sales proposals. Ad hoc workflows, on the other hand, have a low business value and include such activities as interview scheduling. Both, ad hoc and collaborative workflows are supported by groupware. Administrative workflows and production workflows both have a high repetitiveness and therefore, workflow types can be defined for them. Administrative workflows, which have a low business value, include processes in administrative domains, such as routing a travel request or processing a purchase order. Production workflows have a high business value and support an organizations core business, such as claims handling in an insurance company or loan application processing in a bank. Administrative and production workflows are supported by WFMS.

3.2 Workflow modeling

Several approaches proposed in the literature to classify workflow metamodels are reviewed. In [GHS95], existing workflow metamodels are broadly classified into communication-based metamodels and activity-based metamodels, where most of the existing WFMS comprise an activity based metamodel.

Workflow modeling, i.e., creating a workflow type, requires a workflow metamodel that comprises a set of modeling concepts. A concrete workflow type is expressed in a workflow modeling language, which is a formal language offering constructs for the modeling concepts of the metamodel.

See Figure 3.2 for a summary in the Unified Modeling Language (UML) of the relationships that exist among various concepts related to workflow modeling.

A workflow modeling language provides concrete constructs for the concepts of its underlying metamodel. Workflow modeling languages can be broadly classified into textual and graphical modeling

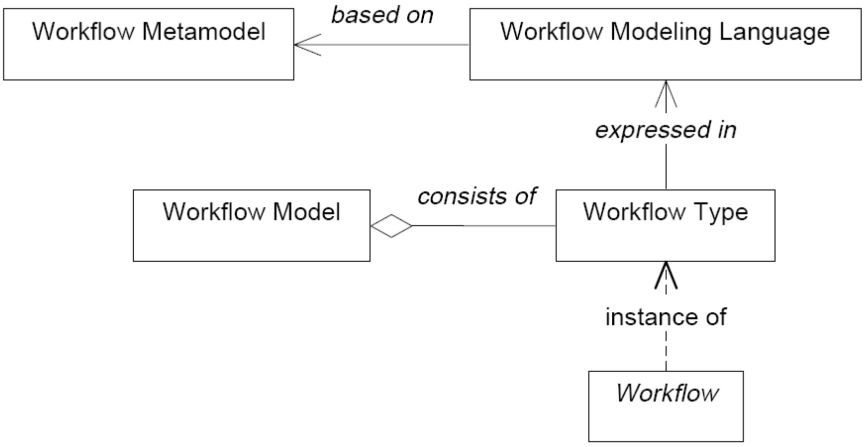


Figure 3.2: Workflow Modeling in UML

languages. Textual workflow modeling languages allow to represent workflow types in a textual way. Graphical workflow modeling languages, on the other hand, allow to visualize certain aspects of workflow types. Typically, these languages allow to represent workflow types as graphs with nodes representing workflows and edges, which connect the nodes, representing various dependencies (e.g., control and data flow) that exist among the workflows. As an example, consider Figure 3.3 that shows a workflow type expressed in a graphical workflow modeling language. In this context it is important to note that graphical workflow modeling languages often also include textual elements in order to express complex information such as conditions.

3.2.1 Document WorkFlow

Before illustrating our approach, we give some remarks on the terminology used. As Document-centric Workflow or Document Workflow (DW) we refer to a particular workflow in which all activities, made by the agents, turn out to documents compilation. It

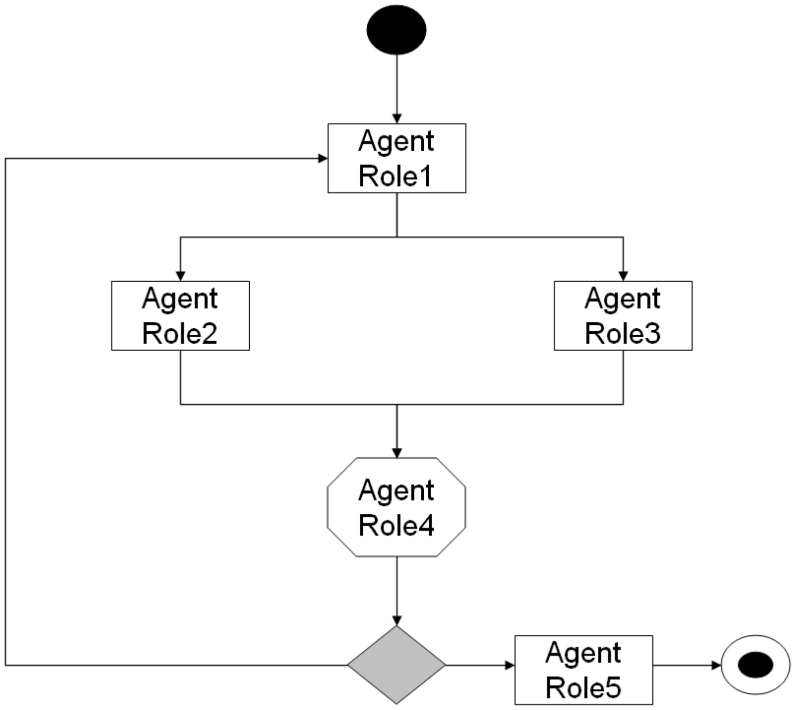


Figure 3.3: A generic document workflow

can be viewed as the automation and administration of particular documents procedures ([AMM01] , [KMK02], [KMK02]). In other words, a DW can be seen as a process of cooperative authoring where the document can be the goal of the process or just a side effect of the cooperation. Through a DW a document life-cycle is tracked and supervised, continually providing document compilation actions control. In this environment a document travels among agents who essentially carry out the pipeline receive-process-send activity.

In our vision there are two types of agents: *external agents* are human or software actors which perform activities dependent from the particular DW, and *internal agents* are software actors providing general-purpose activities useful for any DW and, for this reason, implemented directly into the system. An external agent executes some processing using the document content and eventually other data, updates the document inserting the results of the preceding processing, signs the updating and finally sends the document to the next agent(s). Internal agents perform general functionalities such as creating a document belonging to a particular DW, populating it with some initial data, duplicating a document to send to multiple agents, splitting a document to send partitions to different agents, merging duplicated documents coming from multiple agents, aggregating document fragments, terminating operations on the document. Figure 3.3 illustrates a generic document workflow diagram where external and internal agents cooperate exchanging documents according to some procedural rules.

3.3 Document Workflow Framework

Our document workflow framework is based on *document-centric model* where all the activities, made by the agents, turn out to documents compilation. During its life the document passes through several phases, from its creation to the end of its processing. The state diagram in Figure 3.4 describes the different states of the document instances. At the starting point of document instance life cycle there is a creation phase, in which the system raises a new instance of a document with several information attached (such as the requester agent data). Then document instance goes into pending state. When an agent gets the document, it goes into processing state in which the agent compiles the parts of his competence. If the agent, for some reason, doesn't complete the instance elaboration, he can save the work performed until that moment and the document instance goes into freezing state. If the elaboration is completed (submit), or cancelled, the instance goes back into pending state, waiting for a new elaboration.

In order to automate DWs, an engine with the task of managing all the functionalities to support agent activities is necessary. Our

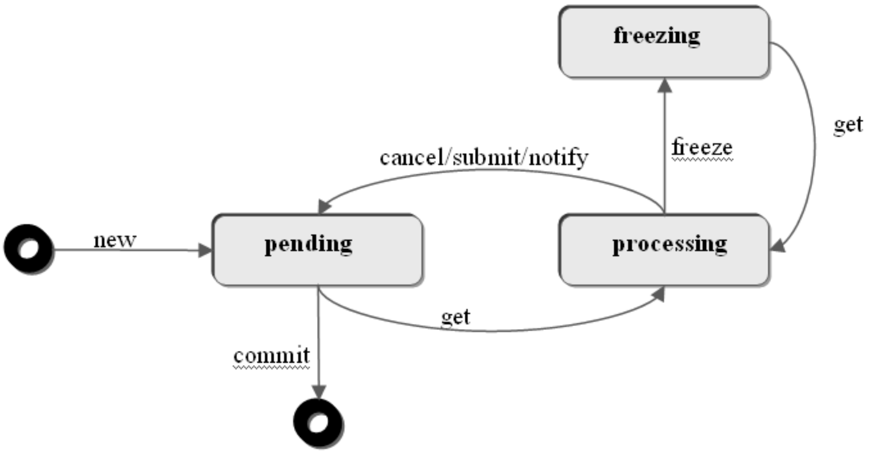


Figure 3.4: State diagram of a document instance.

goal has been to design a *DW engine* which is independent from the single DW. This allows adding new DWs without having to modify the engine. For this reason it's necessary to isolate the information of each DW separating it from the engine. The DW engine will have some parser to interpret these descriptions. We will see in detail the essential components necessary to describe a DW.

- DW Environments (Agents participating to the DW)
- DW Engine
- DW Data (DW descriptions + Documents created by the DW)

3.4 Document Workflow Description

The description of a DW can be seen as an extension of the XML document class. A class of documents, created in a DW, share the schema of their structure, as well as the definition of the procedural rules driving the DW and the list of the agents attending to the DW. Therefore in order to describe a DW we need four components:

- a *schema* of the documents involved in the DW;
- the agent roles chart, called *role chart*, i.e. the set of the external and internal agents, operating on the document flow. Inside the role chart these agents are organized in roles and groups in order to control who accesses the document. This component constitutes the DW environment;
- a *document interface description* used by external agents to access the documents. This component also allows to check the access to the document resource;
- a *document workflow description* defining all the paths that a document can follow in its life-cycle, the activities and policies for each role.

Furthermore the system for keeping track of document instances history (including the agents that have manipulated the document) and document state during its whole flow path needs respectively a Log and Metadata component. The Metadata component represents the document current state. Every time a document changes its state (see Figure 3.4) the Metadata is first saved into Log component and then updated. These last two documents are produced automatically by the DW engine. For each component we define a declarative language, using XML. Hence, each document belonging to a DW will have associated six documents that take for all its life-cycle as indicated in Figure 3.5.

3.4.1 Document Schema

Document schema describes the structure and the data-types of the documents participating to the flow. Document schema will be described using XML Schema [HST]. Typically in a document-based workflow, a document goes through a number of iterations as different people add to its content but it is difficult to design a

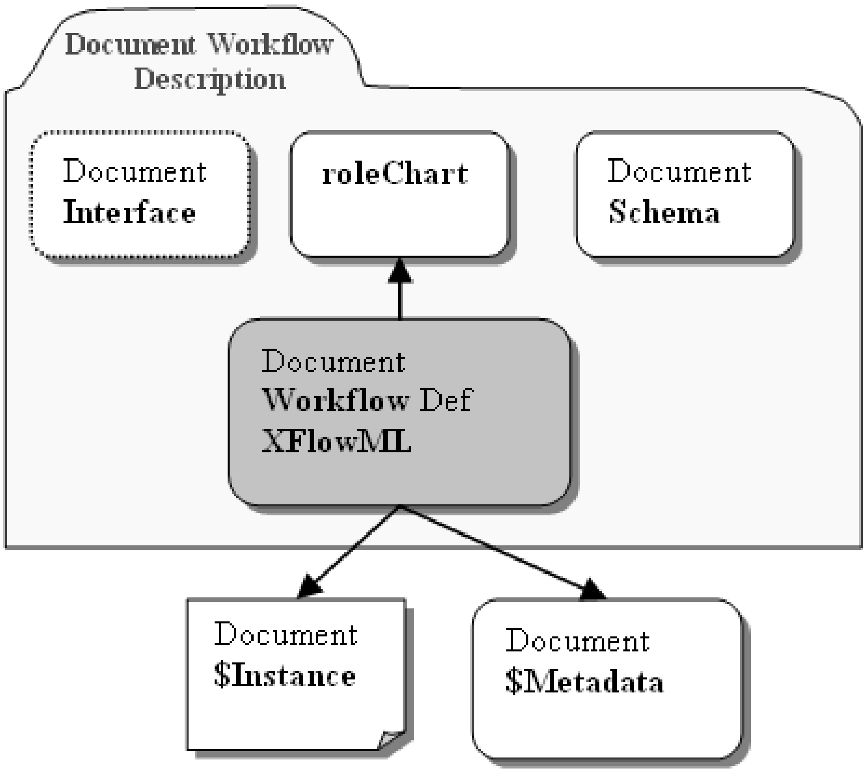


Figure 3.5: Document Centric Vision

schema that describes the document at every possible stage of its lifecycle. Another solution could be to apply different schemas with different sets of rules at each stage of the life-cycle, using validity against a particular schema as the criterion to allow the document to progress to the next stage.

XML Schemas is very powerful but there are many constraints which cannot be expressed with XML Schemas. W3C XML Schema cares not only about validating the structure of XML documents, but also about validating the content of text nodes and attributes and checking the integrity between keys and references. More importantly, W3C XML Schema addresses many issues beyond validation. It attempts to be a modeling language that can classify the elements and attributes of XML documents, identify their semantics, use these semantics as extensible object-like models, and perform automatic binding between XML documents and objects.

In practice, another solution could be to use a very permissive schema that it is capable of describing the document at any stage in its life-cycle, and then to validate for specific stages with a different technology, for example using Schematron [Sch] or a RelaxNG [JC].

3.4.2 Document Interface

This document describes for each agent role the interface toward the document. The document interface for external human agents relies upon Web Modules technologies [JMB06], and external software agents make use of Web Services technologies [DB06], [Mit03]. In the first solutions we adopted XForms technology, promoted by W3C.

3.4.3 Role Chart: Agent Role Declaration

The role chart is an XML document containing the description of all actors (agents) that participate to the workflow. Each actor has a role and a unique identifier. Roles are organized in the role chart hierarchically. Each agent can participate to the workflow with one or more roles, therefore it can appear in one or more role chart positions. The role chart schema is depicted in Figure 3.6.

Finally the document workflow description is a document based on a new XML application (XFlowML Xml document workFlow Markup Language) suitably defined for this purpose.

3.4.4 Document workflow definition

For the definition of a language to describe complex document flows we analyzed several syntaxes and approaches. A possible solution was to use a notation similar to concurrent languages, using statements like *fork* and *join* to describe flows. Another choice was to

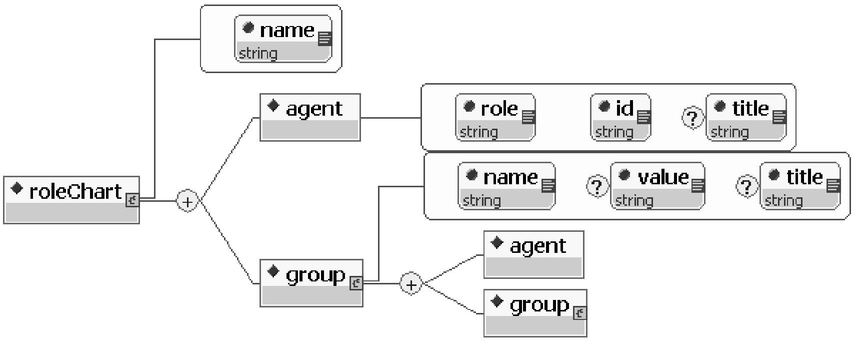


Figure 3.6: RoleChart Schema

describe the document flow from the point of view of the agents. To describe a document flow it is sufficient to accurately describe all the agents and all the operations any agent can perform on the document instance. This way of describing the flow resembles XSL syntax [Kay07], where actions performed by various agents are similar to the templates to apply to the elements of an XML document. Our basic decision to represent flows as XML documents, led us to choose the second option, since with XML, due to its intrinsic hierarchical notation, it is more straightforward to represent lists rather than graphs (other approaches which emphasize the role of XML can be found in [ACLG02], [CTZ02], [AT00], [TS01], [Tol02]). Taking as a simple example the generic flow depicted in Figure 3.3, we will have to supply as many descriptions as the agents roles involved in the process. For instance, in the description of the external agent with role1 (Ag. Role1), we must specify that it can receive documents from Creator or Ag.Role4, and send it to Ag.Role2 and Ag.Role3. To describe document flow we adopted a XML dialect, called XFlowML, largely based-on XSL-Syntax, whose schema is represented in Figure 3.7.

A XFlowML document is composed of a list of internal or external agent. Each agent has a mandatory attribute role, containing a

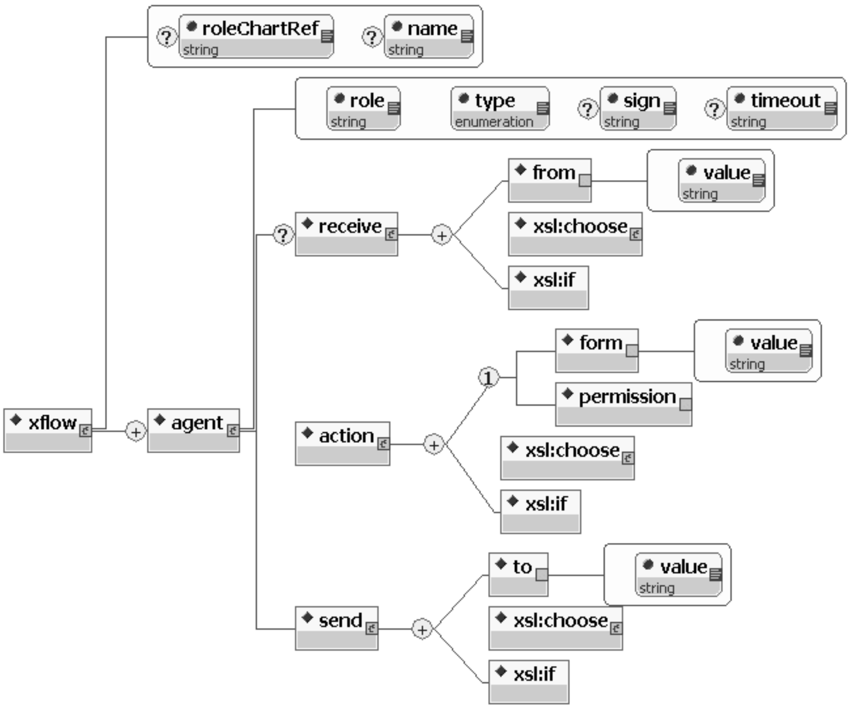


Figure 3.7: XFlowML Schema

XPath expression [JC99] referring to the rolechart. Other optional attributes specify if the agent has to sign the document (sign), and the maximum time the agent is allowed to keep the document (timeout). When an agent requests a document, the DW Engine matches the agent's role on XFlow document and processes the three section: receive, action and send. In the receive section the from elements identify from which agent roles the document can be received. The roles of the agents are coded as XPath expressions. The receive section is optional because it's necessary only to verify if the agent can really receive the current document. In the action section there are one or more permission elements defining the access policies to the document fields. The send section contains all the possible receivers of the document. The document can be sent simultaneously to several agents by using a sequence of to elements. In order to increase the flexibility and power of the language, thus allowing for an easy definition of more complex DWs, we introduced the conditional 'if' and 'choose' statements which adopt the XSLT syntax [Kay07] and can be specified in any agent section (i.e. receive, action, send). Test attributes can contain any XPath expression which returns a Boolean, and it is possible to refer document Metadata or document instance. For distinguishing the referred document the test XPath expression will begin with two different prefixes: respectively $Metadata and $Instance.

Inside XFlowML document we can have XPath expression referring elements or attribute of Rolechart, Document, and Metadata. To distinguish them, they have the prefixes $Instance e $Metadata. A typical use of these conditional statements is in send element, when we have to send the document to two different agent depending on the value of a document field previously filled out.

```
<send>
  <xsl:choose >
   <xsl:when test="$instance//agent[approved='true']">
     <to select=//agent[@role='manager']
   <xsl:when>
   <xsl:otherwise>
     <to value="//agent[@role='employee']">
   <xsl:otherwise>
  </xsl:choose>
</send>
```

3.4.5 Metadata and Log Components

For each document instance that participate to a specific DW, additional information (that we have called metadata) is stored together with information to reconstruct the document history (Log) that consists of all the document transitions during its processing, including also information about actors involved. Log permits to undo actions. The metadata associated with each document is composed by following fields.

- **urn**: Univocal document's name. It doesn't change from its creation until its registration.
- flowId: Identifiers of the flow who the instance belong to. It doesn't change from its creation until its registration.
- docTitle: Document's title. It's depended from flow.
- docFileName: Instance file's name. It doesn't change from its creation until its registration.
- timestamp: Date of the creation of the tupla. It corresponds to the state change of the document.
- creator: RolePathId of the agent who has instanced the document. It doesn't change from its creation until its registration.
- sender: RolePathId of the agent who has done a submit on instance.
- receiver: rolePathId of the agent who have to receive the instance. It can be a rolePathId to identify exactly an agent or a rolePath to identify a group of agents that can receive without distinct the document.
- owner: agentId of the agent who is elaborating the instance.
- status: State of the instance (processing, frozen, pending archived)

3.5 The Document Workflow Engine

The document workflow engine constitutes the run-time support for the DW, it implements the internal agents, the support for agent's activities, and some system modules that the external agents have to use to interact with the DW system. Also, the engine is responsible for two kinds of documents useful for each document flow: the documents system logs and the document system metadata.

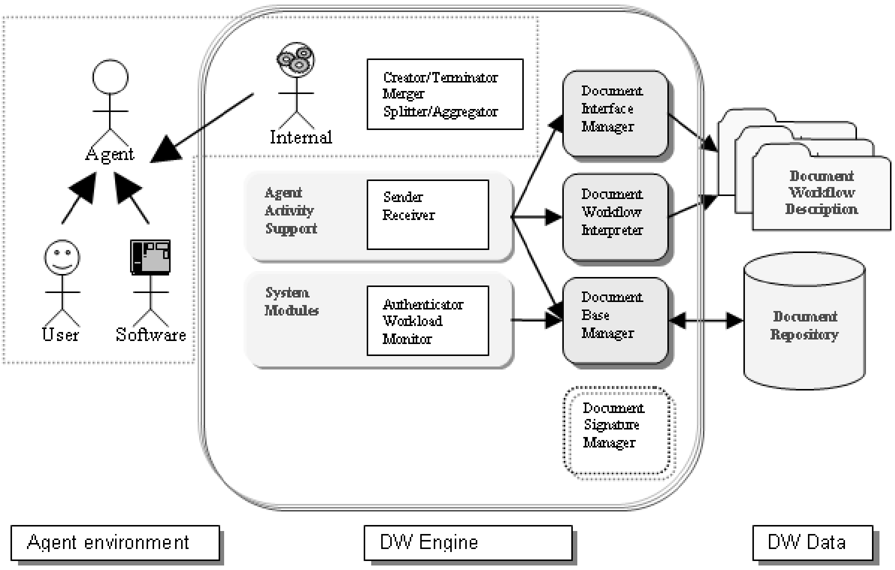


Figure 3.8: Document Workflow Framework

3.5.1 Agent's activities support

These are the two modules, called Sender and Receiver, supporting the activities of sending to, and receiving from the current agent . The Sender has to prepare and send the document (identified by an URN) requested by an external agent. It checks the agent rights, verifies if the document instance is still available, analyzes/interprets the workflow description to generate an adapted document, using the agent's role and access rights to determine which are the parts of the stored document to be included. (using XForms for a human agent and SOAP for a software agent). The Receiver gets the document from the handling agent in consequence of a submitt, freeze or cancel command. It determines the roles of the next agents to whom the document must be sent. Both modules use the DW Interpreter which transforms the XFlow document into a XSLT stylesheet. The generated stylesheet is applied to the Rolechart document producing the role agent's activities.

3.6 Implementation overview

3.6.1 The client side: external agent interaction

Our system is currently implemented as a web-based application where the human external agents interact with system through a web browser. All the human external agents attending the different document workflows are the users of system. In Figure 3.9 and 3.10 the use cases and the state diagram of user activities are shown.

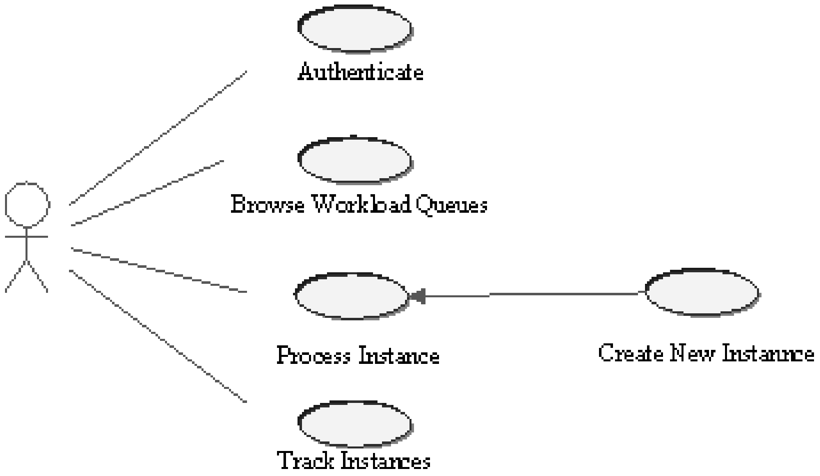


Figure 3.9: useCasesXFlow

Once authenticated through user/psw (Figure 3.11A) the user accesses his workload area (Figure 3.11B) where the system lists all his pending documents sorted by flow. The system shows only the flows to which the user has access. From the workload area the user can browse his/her documents and select some operations such as:

1. select and process a pending document (Figure 3.11C)

2. create a new document partly filled with his/her data.

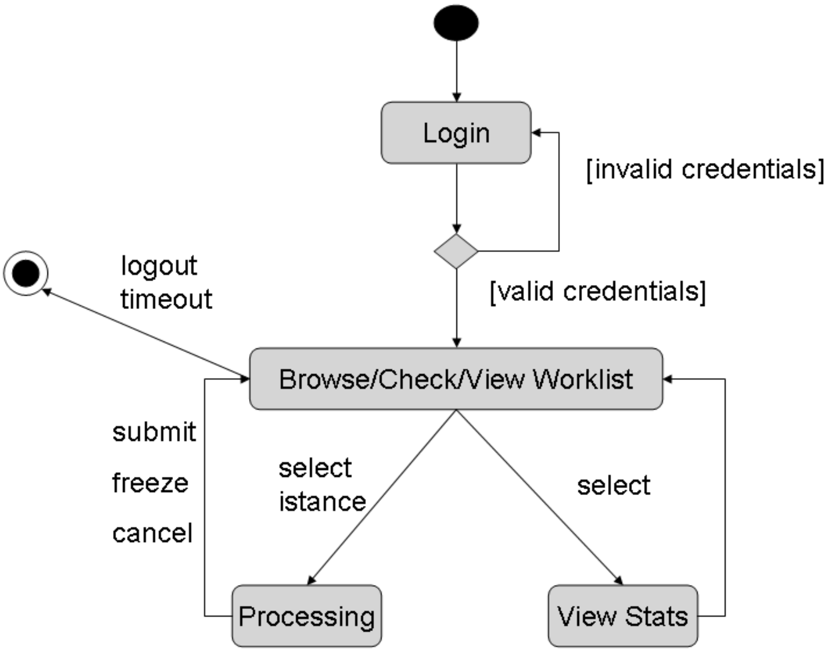
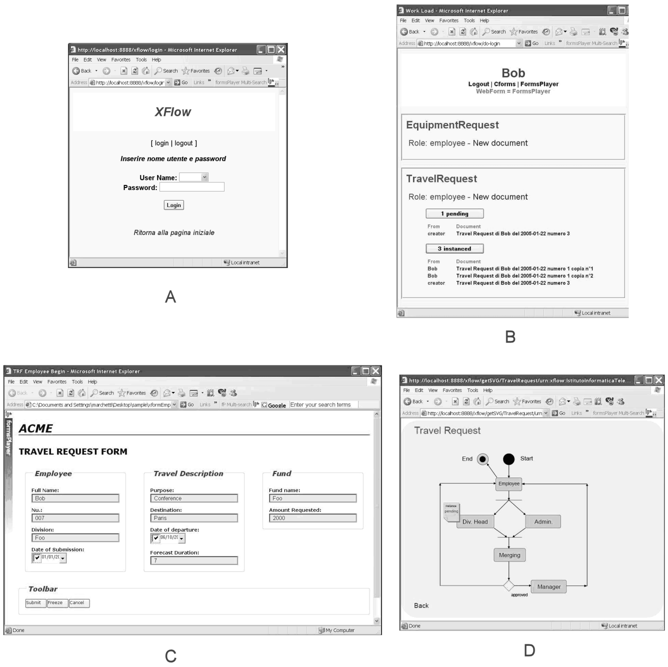


Figure 3.10: stateDiagramXFlow

3. display a graph representing a DW of a previously created document, highlighting the current position of the document (Figure 3.11D). This information is rendered as an SVG image.

The form used to process the documents is rendered with XForms [JMB06] (Figure 3.11C). XForms can communicate with the server by means of XML documents and is capable of displaying the document with a user interface that can be defined for each type of document. XForms is a recommendation of the W3C for the specification of Web forms. In XForms the description of how the form is displayed is separated from the description of what the form must do, so it is easy to use different type of views depending on the platform and on the document. A browser with XForms capabilities



http://localhost:8888/xflow/login - Microsoft Internet Explorer

XFlow

[login | logout]

Inserire nome utente e password

User Name:
Password:

Ritorna alla pagina iniziale

Work Load - Microsoft Internet Explorer

Address: http://localhost:8888/xflow/do-login

Bob

Logout | Cforms | FormsPlayer
WebForm = FormsPlayer

EquipmentRequest

Role: employee - New document

TravelRequest

Role: employee - New document

1 pending

From	Document
creator	Travel Request di Bob del 2005-01-22 numero 3

3 instanced

From	Document
Bob	Travel Request di Bob del 2005-01-22 numero 1 copia n°1
Bob	Travel Request di Bob del 2005-01-22 numero 1 copia n°2
creator	Travel Request di Bob del 2005-01-22 numero 3

TRF Employee Begin - Microsoft Internet Explorer

Address: C:\Documents and Settings\marchetti\Desktop\sample\xformEmp

ACME

TRAVEL REQUEST FORM

Employee
Full Name: Bob
Nu.: 007
Division: Foo
Date of Submission: 01/01/20

Travel Description
Purpose: Conference
Destination: Paris
Date of departure: 06/10/20
Forecast Duration: 7

Fund
Fund name: Foo
Amount Requested: 2000

Toolbar

http://localhost:8888/xflow/getSVG/TravelRequest/urn:xflow:IstitutoInformaticaTele...

Address: http://localhost:8888/xflow/getSVG/TravelRequest/urn

Travel Request

Figure 3.11: screenShots

will receive an XML document that will be displayed according to the specified template, then it will let the user edit the document and finally it will send the modified document to the server.

The server-side is implemented with Apache Tomcat, Apache Cocoon and MySql. Tomcat is used as the web server, authentication module (when the communication between the server and the client needs to be encrypted) and servlet container. Cocoon is a publishing framework that uses the power of XML.The entire functioning of Cocoon is based on one key concept: component pipelines. The pipeline connotes a series of events, which consists of taking a request as input, processing and transforming it, and then giving the desired response. The pipeline components are generators, transformers, and serializers. A Generator is used to create an XML structure from an input source (file, directory, stream ...)

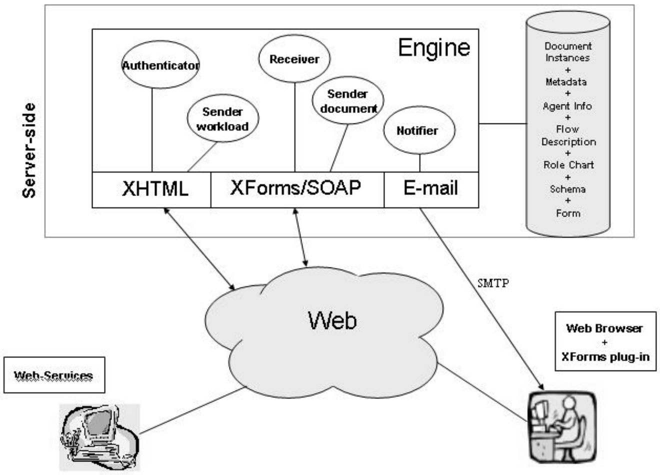

Server-side
Engine
Authenticator
Receiver
Sender workload
Sender document
Notifier
XHTML
XForms/SOAP
E-mail
Document Instances + Metadata + Agent Info + Flow Description + Role Chart + Schema + Form
SMTP
Web
Web Browser + XForms plug-in
Web-Services

Figure 3.12: XFlowImplementation

A Transformer is used to map an input XML structure into another XML structure (the most used is XSLT transformer). A Serializer is used to render an input XML structure into some other format (not necessarily XML) MySql is used for storing and retrieving the documents and the status of the documents. There are some modules that allow the interaction with the user agents. The Authenticator and WorkloadSender modules use XHTML to display data. The Receive and Sender Document modules use XForms to exchange XML document with human agents and the SOAP protocol to exchange documents with software agents. (Figure 3.12).

The benefits arising from the usage of this system include:

- interoperability/portability deriving from using XML technologies;
- concurrent documents workflows managing;
- reduction of the cost of documents processes, through the e-documents processing and distribution;

Finally, XML is a suitable technology for representing not only documents but also to describe the document workflow logic. XFlowML is the XML application defined to describe a document workflow. We have defined a model to describe DW based on three XML documents (schema, rolechart and xflow) that allows an easy description of many DW. We have implemented a DW engine interpreting XFlowML documents, by using Cocoon, a very powerful middleware, to develop XML prototypes. The DW engine implemented is heavily based on XML technologies (XSLT, XPath, XForms, SVG) and open-source tools (Cocoon, Tomcat, mySQL). We have used Xforms technology to create dynamic user interfaces.

Chapter 4

LexFlow

Abstract

In the last years many scholars have called for a new generation of semantic resources, where content is dynamically augmented by resorting to heterogeneous sources. In order to attain better coverage, it should be possible for semantic resources to be automatically maintained and augmented, possibly by resorting to sources other than human, introspective knowledge and integrating the knowledge either already explicitly encoded in other resources, or implicitly conveyed by corpora. This chapter presents LeXFlow, a framework for the semi-automatic management of lexical entries. LeXFlow is intended to provide an architectural and practical framework enabling dynamic, semi-automatic integration of semantic resources, exemplifying the particular case of semantic computational lexicons. In this chapter, we describe LeXFlow, a metaphoric extension and adaptation of XFlow, and we present a sample lexical flow corresponding to the scenario where an entry of a lexicon becomes enriched with semantic relations available in an other lexicon and informations come from corpora. The relations acquired are integrated into the entry and proposed to the human encoder for final checking and validation.

4.1 Moving to dynamic computational lexicons

Computational lexicons aim at providing an explicit representation of word meaning, so that it can be directly accessed and used by computational agents. In the last decade, many activities at European level and worldwide have contributed to substantially advance knowledge and capability of how to represent, create, maintain, acquire, access, and share large lexical repositories. However, most existing lexical resources do not have enough coverage, not only for practical reasons, but also for more structural and inherent reasons. No individual static resource can ever be adequate and satisfying, neither in extension (since it cannot cover new formations, or all the possible domains) nor in depth (since it cannot provide all the necessary and useful linguistic information, not even for the existing lexical entries). The computational lexicon community is thus increasingly calling for a change in perspective on computational lexicons: from static resources towards dynamic multi-source entities, integrating and harmonizing the linguistic information coming from different sources, where lexical content is co-determined by automatically acquired linguistic information from text corpora and from the web. A different scenario is thus envisaged, where acquisition tools are able to increase the repository with new words/terms, possibly their definitions, domain, etc., from digital material, to learn concepts from text, and to tailor resources to specific needs.

We believe that an essential step towards the realization of the dynamic paradigm of lexical resources is closely related to the development of an appropriate framework for computational lexicons where lexical entries behave as semi-independent entities, that dynamically modify and update their content on the basis of the integration of knowledge coming from different sources, where the sources can be indifferently represented by human agents, other lexical resources, or applications for the automatic extraction of lexical information from texts. This scenario has at least two strictly related prerequisites: on the one hand, it assumes that existing lexicons are available in a form enabling the overcoming of their respective differences and idiosyncrasies, thus making their mutual comprehensibility a reality. On the other, it calls for the provision of an architectural framework for the effective and practical management of lexicons, by providing the communicative channel through which lexicons can really communicate and share the information encoded therein.

4.1.1 General architecture: the metaphor of lexical workflow

Similarly to document workflow management system, the management of computational lexicons can be described as a flow of lexical entries. A lexical entry is modeled as a document moving through different agents, with clear-cut roles, acting over different portions of each entry. Following this metaphor, LeXFlow is conceived as a metaphoric extension and adaptation to computational lexicons of XFlow, a framework for the collaborative management of document workflows [MTM05].

In this environment there are two types of agents: internal agents are software actors providing general-purpose activities useful for any workflow and hence are implemented directly into the system, while external agents are human or software actors that perform activities dependent from a particular lexical workflow (LW). Internal agents perform general functionalities such as creating/converting an entry belonging to a particular LW, populating it with some initial data, duplicating an entry to be sent to multiple agents, splitting an entry and sending portions of information to different agents, merging duplicated entries coming from multiple agents, aggregating fragments, and finally terminating operations over the entry. External agents basically execute some processing using the already available content of the entry and populate it with lexical information. In our demonstrative LW, a particular type of external agent is represented by an application that acquires information about part-of relations by identifying syntactic constructions that are often used to express such relations, in a vein similar to [MPV02]. Other external agents are one or more compilers and one or more roles for quality control, who basically check the output of the previous agent(s), validate it, and send the document to the next agent(s) (see Figure 4.2 below). In order to account for the peculiarities of lexicon encoding and management, XFlow has been extended and specialized. In the LeXFlow framework the workflow of lexical entries is described by a new XML application called XFlowML (XFlow Markup Language), largely based on XSLT Processing Model. XFlowML describes a workflow using an agent-based approach. Each human or software agent can participate to the workflow with one or more roles, defined as XPath expressions, based on a hierarchical role chart. An XFlowML document contains as many templates as are the agent roles participating in the workflow. The selection of the templates will establish the order with which the agents will receive the lexical entry. The document workflow engine constitutes the runtime execution support for the document processing by implementing the XFlowML constructs. To this end, at first we have defined the logical schema of a lexical entry and the contextual domain of the document workflow including all human and software agents cooperating, with different roles, to the compilation of lexical entries. Finally we have formalized the procedural rules and the access control rules (XFlowML) of lexical entry compilation. A prototype of LeXFlow has been implemented with an extensive use of XML technologies (XML Schema, XSLT, XPath, XForms, SVG) and open-source tools (Cocoon, Tomcat, mySQL). It is a web-based application where human agents interact with the system through an XForms browser that displays the document to process as a web form whereas software agents interact with the system via web services.

4.1.2 Representing lexical entries: the MILE lexical model

In order to ensure interoperability, an essential prerequisite is the requirement that lexicon entries be encoded in a shared, standard format. We have chosen to use the MILE [NC03] as a standardized model to describe the entries belonging to different lexicons. The

MILE is a general architecture devised for the encoding of multilingual lexical information, a meta-entry acting as a common representational layer for multilingual lexicons, by allowing integration and interoperability between different monolingual lexicons. Although primarily devised for multilingual lexicons, the MILE can also be applied to mono-lingual lexicons. MILE-conformant lexical entries can be built by lexicon and application developers by means of the overall MILE Lexical Model (MLM). According to the model, the monolingual component on the vertical dimension is organized over three different representational layers which allow to describe different dimensions of lexical entries, namely the morphological, syntactic and semantic layers. Moreover, an intermediate module allows to define mechanisms of linkage and mapping between the syntactic and semantic layers.

Within each layer, a basic linguistic information unit is identified; basic units are separated but still interlinked each other across the different layers. The basic conceptual components of the MILE lexical model are the following:

1. the MILE Lexical Classes (MLC) represent the main building blocks which formalize the basic lexical notions. They can be seen as a set of structural elements organized in a layered fashion: they constitute an ontology of lexical objects as an abstraction over different lexical models and architectures. These elements are the backbone of the structural model. These include main syntactic constructions, basic operations and conditions to establish multilingual links, macro-semantic objects, such as lexical conceptual templates acting as general constraints for the encoding of semantic units.

2. the MILE Lexical Data Categories (MDC) which constitute the attributes and values to adorn the structural classes and allow concrete entries to be instantiated. Typical instances of MDCs are syntactic and semantic features, semantic relations, syntactic constructions, predicates and arguments etc.

the MILE Lexical Data Categories (MDC) which constitute the attributes and values to adorn the structural classes and allow concrete entries to be instantiated. Typical instances of MDCs are syntactic and semantic features, semantic relations, syntactic constructions, predicates and arguments etc. MILE appears especially suited to our needs by virtue of being a) modular (different levels independently encoded), and b) granular (different degrees of depth at which an entry can be described at each level). Since our case study concerns the semantic information of a lexical entry, we will concentrate on the semantic layer only, as illustrated in Figure 4.1.

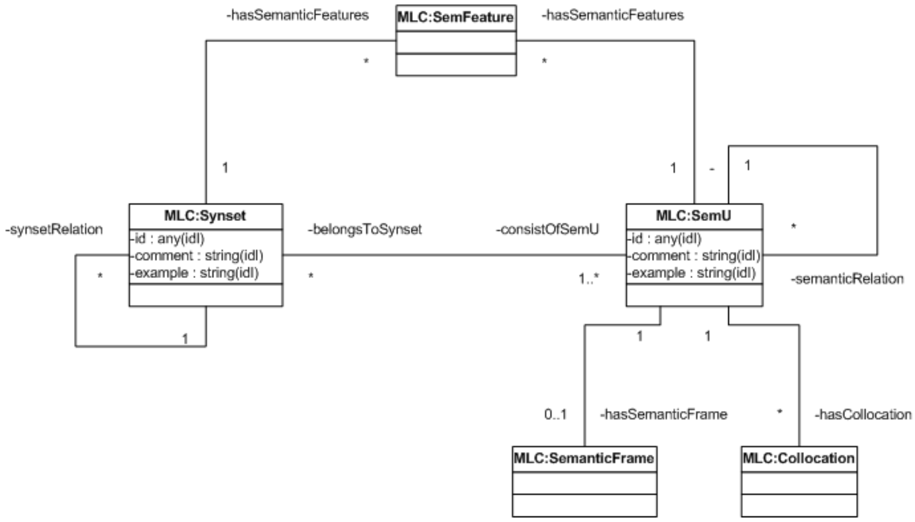


Figure 4.1: MILE

Originally, in order to meet expectations placed upon lexicons as critical resources for content processing in the Semantic Web, the MILE syntactic and semantic lexical objects have been formalized in RDF(S), thus providing a web-based means to implement the MILE architecture and allowing for encoding individual lexical entries as instances of the model [ILC03]. In the framework of our project, by situating our work in the context of W3C standards and relying on standardized technologies underlying this community, the original RDF schema for ISLE lexical entries has been made compliant to OWL.

4.2 Integrating lexicons using LeXFlow

LeXFlow is not to be intended as a tool for the compilation or editing of lexicons (although it can be used to such an end). While different flows can be envisaged, depending on the particular needs as well as on the particular attitude towards the work of a lexicographer, we demonstrate the potential of LeXFlow by illustrating its application to the case where two different semantic lexicons interact by reciprocally enriching themselves and integrating information coming from corpora. To this end, we have designed a sample lexical flow (see Figure 4.2) corresponding to the scenario where an entry of a lexicon A becomes enriched via basically two steps. First, by virtue of being mapped onto a corresponding entry belonging to lexicon B, the entry inherits the semantic relations available in lexicon B, and vice-versa. Second, by resorting to an automatic application that acquires information about semantic relations from corpora, the relations acquired are integrated into the entry and proposed to the human encoder for final checking and validation.

The aim of this lexical flow is thus threefold:

- to enrich the entries of a lexicon with information coming from corpora and from a foreign lexicon;
- to show how the MILE lexical model not only allows, but enforces the integration;
- to provide an instrument, based on the MILE model, that allows the creation of enriched lexical entries, where the information coming from different lexicons is fused.

For our purposes, we chose to enrich the ItalWordNet [RA03] and the SIMPLE/CLIPS [RN03] lexicons. These two semantic lexicons represent two very different attitudes towards the description of semantic content, and hence encode different types of information. In our scenario, it is assumed that the two lexicons are already represented according to the MILE specifications. We recall that, according to the MILE model, an entry coincides with a given sense of a word (a SemU, Semantic Unit). In the simplified MILE-conformant entry schema we have adopted, each SemU is encoded as a single document A SemU is described by means of the following attributes:

- an ID
- a gloss
- the lemma
- an example
- an indication of the source

For the sake of readability, moreover, we overtly simplified the complexity of the two lexicon encodings by concentrating only on a subset of the range of semantic information available and currently encoded in lexicons. In particular, we decided to focus on the bunch of semantic relations (hyponymy, synonymy, meronymy, and the like) that a given sense of a lexical entry has with other senses of the same lexicon. Thus, for the SIMPLE/CLIPS lexicon, each SemU is further described by means of a list of semantic relations, each of them linked to a target SemU. On the other hand, in the MILE-conformant version of the ItalWordNet lexicon, each SemU corresponds to a variant of a given synset. Apart from the general descriptive fields described above, a wordnet-derived SemU only contains indication of the native synset, a notion expressed by the belongsToSynset relation. The semantic relations describing the relational context of a variant are described inside the synset. In the following subsections we give a step-by-step description of the flow, whose overall picture is represented in Figure 4.2. The Figure clearly illustrates the different agents participating to the flow. Rectangles represent human actors over the entries, while the other Figures symbolize software agents: ovals are internal agents and octagons external ones.

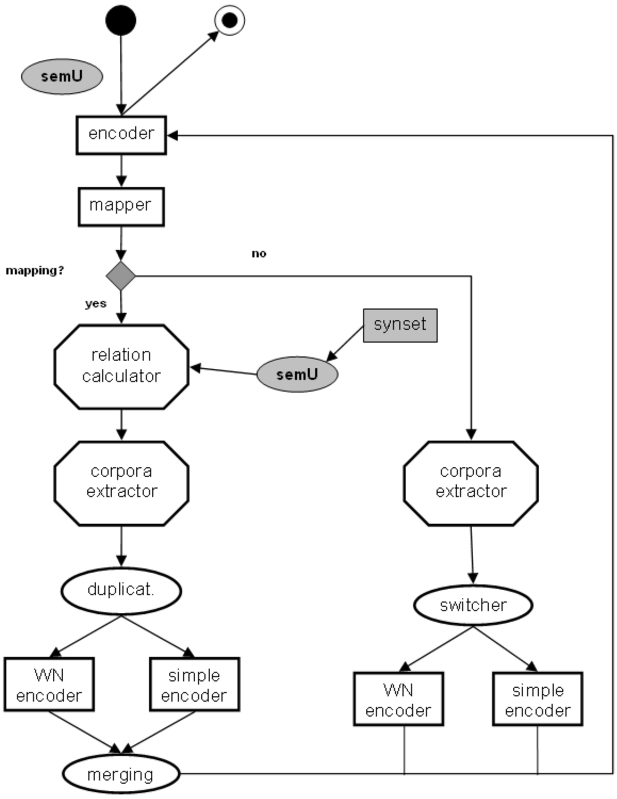


Figure 4.2: Lexical flow activity diagram

4.2.1 Starting the flow: the mapping phase

In this scenario, a user or encoder starts by selecting an entry of a semantic lexicon that will represent the instance to be processed by the flow. Suppose that the selected entry is the SemU *car1*, belonging to the SIMPLE/CLIPS lexicon. After this first step, the entry becomes processed by another user, having the role of mapper. The mapper selects a corresponding entry belonging to the ItalWordNet lexicon that expresses the same sense. Lets assume that the mapper has identified a corresponding entry in the SemU *car2* belonging to the Synset *car2auto1machine4* of the ItalWordNet lexicon. For the sake of simplicity we hypothesize a human agent, but the same role could be performed by a software agent.

4.2.2 Merging of semantic relations

If the mapping procedure is successful, then the two instances (entries) are loaded and aggregated in a single object. At this stage, this new object includes all the relations originally pertaining to the originating instances. That is, in this new object there will be the semantic relations as expressed in the SIMPLE-CLIPS lexicon as well as the Synset Relations as expressed in the IWN lexicon. The two different types of semantic relations will target the original targets, that is, the original SemUs for the SIMPLE lexicon and the original synsets for the IWN lexicon.

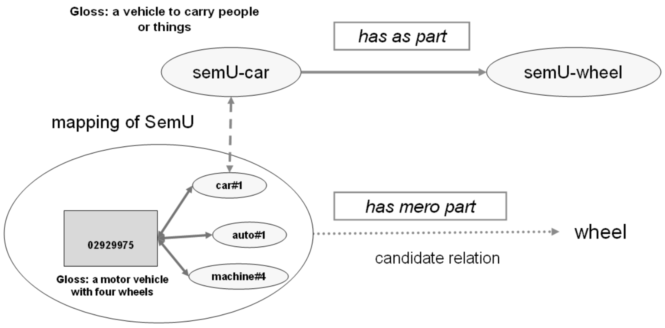


Figure 4.3: Candidate synset relations

The following step is represented by the relation calculator. This software agent is responsible for creating for each lexicon a set of candidate relations on the basis of those available in the other lexicon. It does so by performing two operations: first, it translates the semantic relations coming from a lexicon into the parlance of the other lexicon. Second, it creates for the imported relations as many candidate targets as are the original targets (either SemUs or Synsets).

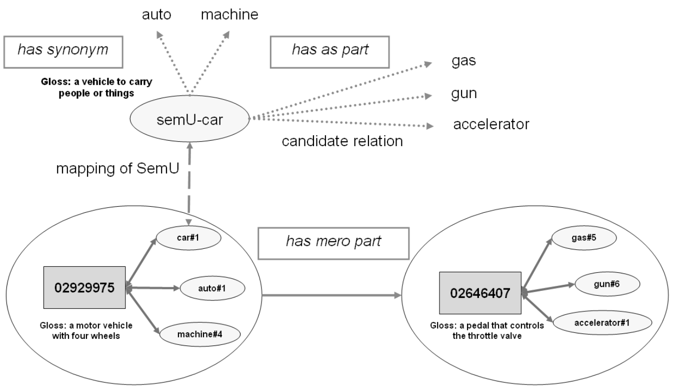


Figure 4.4: Candidate semantic relations

For instance, let's suppose that the SIMPLE entry for *car1* has a hasaspart semantic relation with another entry, namely *wheel1*. Figure 4.3 illustrates this scenario. The Relation Calculator then creates a translation of each semantic relation into the language of the other lexicon, to be proposed for validation in a subsequent step. In the case at hand, it will translate the hasaspart relation into the corresponding hasmeropart synset relation. The targets of these candidate relations will not be SemU, but the procedure will propose a candidate lemma for each relation. It will be the encoders duty to associate a proper SemU belonging to his lexicon to a candidate relation. Moreover, if the SIMPLE-derived SemU contains some has-synonym relations then LeXFlow proposes a widening of the IWN synset by means of the lemma corresponding to the target SemU. On the other hand, for each synset relation encoded for a WordNet-derived SemU for which there is an equivalent relation in the SIMPLE parlance, LeXFlow proposes as many candidate semantic relations as the SemUs contained in the target synset (see Figure 4.4). Once again, every candidate semantic relation points to a lemma. In addition, LeXFlow creates as many semantic relations of the hassynonym type as are the variants belonging to the IWN corresponding synset. The Relation Calculator simply ignores those relations that cannot be mapped.

4.2.3 Automatic acquisition from corpora

At this stage, the instance representing the unit under processing by the flow has been enriched with a set of potential semantic relations, as a result of the crossbreeding between the corresponding entries as encoded in the two source lexicons. The following step is represented by the action of an application that acquires information about part-of relations by identifying syntactic constructions in a vast Italian corpus of about 90 million words [MR03]. The corpus was previously analysed by Chunk-It [LA03], a chunker developed at ILC-CNR as part of a complete chain for the linguistic analysis of Italian. The flow invokes the application by sending a query on the basis of the lemma of the entry under processing. The application essentially consists in a grammar whose rules are syntactic patterns that can be indicative of meronymy relations. The output of the automatic procedure is then acquired by LeXFlow, that takes care of creating the appropriate candidate semantic and synset relations for each lemma that is proposed by the application. A lemma is automatically discarded as a candidate target if it is already present as target of a semantic or synset relation in the list of those already encoded in the entry (either originally or as a result of the merging step).

4.2.4 Enrichment of semantic relations

After these steps, LeXFlow duplicates the instance and sends it to two human agents, identified as a SIMPLEencoder and an IWN encoder. Their duty consists in accepting or discarding the proposed relations, as well as choosing the appropriate target SemUs or Synset for each relation that is proposed. It is worth noting that LeXFlow produces two separate views of the same enriched entry by showing only the portions that are relevant to the different starting lexicons. In other words, the SIMPLE encoder will be able to validate only the semantic relations already translated into the SIMPLE parlance. On the other hand, these will remain opaque to the IWN encoder, who will check the proposed Synset relations only. In each separate view, LeXFlow provides to the encoders a window where starting from the proposed target lemmas the user can either choose the target SemU or Synset from the original lexicons (if already available), or either creating it from scratch.

4.2.5 Ending the flow

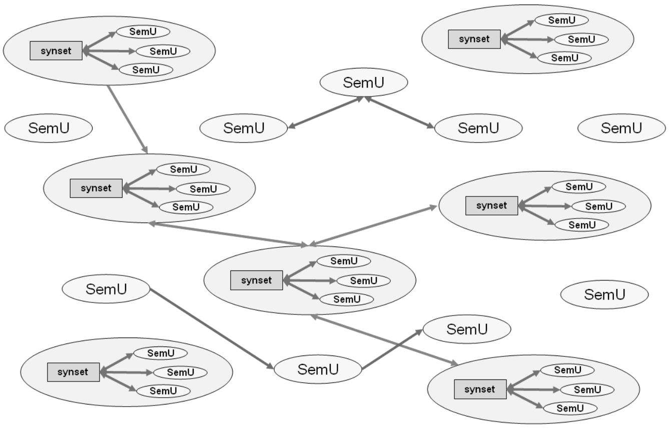


Figure 4.5: Lexicon initial state

After the validation phase, the flow again makes a merging of the two versions of the entry, by joining the portions that have been modified by the two encoders. The merged entry is then returned to the initial user for a final check. If accepted, this new entry replaces the original entry in the lexical database. It is worth noting that the replacement takes place in both lexicons, thus providing a true contamination of the two worlds, although controlled. Since the entry is expressed in the MILE model, that provides the expressive power to allow for different views over the same semantic space, the contamination is not only allowed but enforced, thus paving the way for a truly merged lexicon to be created. In fact, the lexical flow described provides all the means for linking two lexicons in an integrated repository, with all entries opening doors over the two originating worlds. Initially the two lexical repositories are completely separated, although compatible thanks to the interlingua provided by the MILE encoding. This situation is illustrated by Figure 4.5. SIMPLE SemUs and IWN SemUs co-exist into the same space, but are by no means connected, with the former being linked only among themselves, and the latter only living into the restricted space of the Synsets to which they belong. After completing several flows, we gradually arrive at a situation where the two lexicons begin to integrate, with cross-breeded SemUs (participating of the properties of both lexicons) throwing links to IWN synsets and to SIMPLE SemUs (see Figure 4.6).

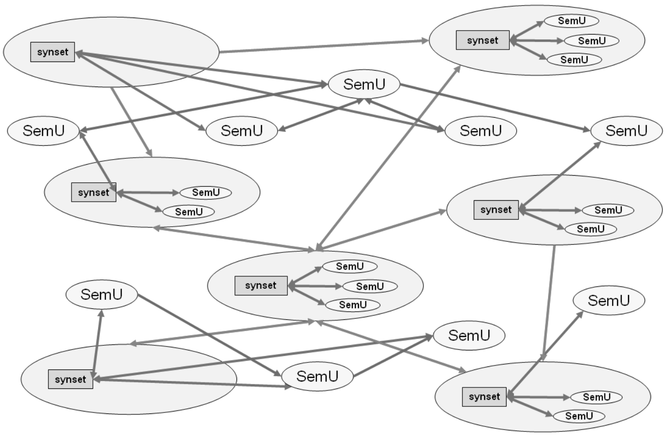


Figure 4.6: Lexicon running state

In this chapter we have illustrated an application of LeXFlow to the merging of different semantic lexicons, with a focus on the enrichment of source lexicons. The same principles can be applied in a scenario where a user is interested in combining different layers of lexical information, for instance phonetic and morphological information [MM06b]. In the flow described in this chapter the outcoming entries enter again into the original lexical repositories, and their merging is almost exclusively exploited in order to enrich their respective set of semantic relations. However, the new entries potentially contain the seeds for representing the building blocks of a truly integrated lexicon, where all the entries are in common. Investigating the possibility of creating a new global lexicon, where each addition or deletion of entries on each side (SIMPLE or IWN) has immediate and automatic consequences on the other represents the commitment of our future work.

Chapter 5

Multilingual WN Service

Abstract

While a number of multi-lingual resources already exist, few of them are practically useful, either since they are not sufficiently broad or because they don't cover the necessary level of detailed information. Moreover, multilingual semantic resources are not so widely available and are very costly to construct: the work process for manual development of new semantic resources or for tailoring existing ones is too expensive in terms of effort and time to be practically attractive. In this chapter we present an application carrying out the integration and interoperability of computational lexicons, focusing on the particular case of mutual linking and cross-lingual enrichment of two wordnets. The development of this application is intended as a case-study and a test-bed for finding needs and requirements posed by the challenge of semi-automatic integration and enrichment of multilingual lexicons. The chapter is organized as follows: section 5.1 describes the general architectural design of our project; section 5.2 describes the module taking care of cross-lingual integration of lexical resources, by also presenting a case-study involving an Italian and Chinese lexicons. Finally, section 5.4 presents our considerations and lessons learned on the basis of this exploratory testing.

5.1 Architecture

The need of ever growing semantic resources for effective multilingual content processing has urged the language resource and semantic web community to call for a radical change in the perspective of semantic resource creation and maintenance and the design of a "new generation" of semantic resources: from static, closed and locally developed resources to shared and distributed semantic services, based on open content interoperability standards. This has often been called a "change in paradigm" (in the sense of Kuhn, see [CN05] [N.06]). Leaving aside the tantalizing task of building on-site resources, the new paradigm depicts a scenario where semantic resources are cooperatively built as the result of controlled cooperation of different agents, adopting the paradigm of accumulation of knowledge so successful in more mature disciplines, such as biology and physics [N.06]. According to this vision, different semantic resources reside over distributed places and can not only be accessed but choreographed by agents presiding the actions that can be executed over them. This implies the ability to build on each other achievements, to merge results, and to have them accessible to various systems and applications. Since language evolves and changes over time, it is not possible to describe the current state of the language away from where the language is spoken. Lastly, the vast range of diversity of languages also makes it impossible to have one single universal centralized resource, or even a centralized repository of resources.

In the Chapter 4 we have illustrated the general architecture of LeXFlow and showed how a Lexical Workflow Type can be implemented in order to enrich already existing lexicons belonging to the same language but realizing different models of lexicon encoding. In this section we move to a cross-lingual perspective of lexicon integration. We present a module that similarly addresses the issue of lexicon augmentation or enrichment focusing on mutual enrichment of two wordnets in different languages and residing at different sites. This module, named "multilingual WN Service" is responsible for the automatic cross-lingual fertilization of lexicons having a WordNet-like structure. Put it very simply, the idea behind this module is that a monolingual wordnet can be enriched by accessing the semantic information encoded in corresponding entries of other monolingual wordnets. Since each entry in the monolingual lexicons is linked to the Interlingual Index (ILI), a synset of a WN(A) is indirectly linked to another synset in another WN(B). On the basis of this correspondence, a synset(A) can be enriched by importing the relations that the corresponding synset(B) holds with other synsets(B), and vice-versa. Moreover, the enrichment of WN(A) will not only import the relations found in WN(B), but it will also propose target synsets in the language(A) on the basis of those found in language(B). The various WN lexicons reside over distributed servers and can be queried through web service interfaces. The overall architecture for multilingual wordnet service is depicted in Figure 5.1.

Put in the framework of the general LeXFlow architecture, the Multilingual wordnet Service can be seen as an additional external software agent that can be added to the augmentation workflow or included in other types of lexical flows. For instance, it can be used not only to enrich a monolingual lexicon but to bootstrap a bilingual lexicon.

5.2 Linking Lexicons through the ILI

The entire mechanism of the Multilingual WN Service is based on the exploitation of Interlingual Index [PVDOA98], an unstructured version of WordNet used in EuroWordNet [Vos04] to link wordnets of different languages; each synset in the language-specific wordnet is linked to at least one record of the ILI by means of a set of equivalence relations (among which the most important is the EQ_SYNONYM, that expresses a total, perfect equivalence between two synsets). Figure 5.2 describes the schema of a WN lexical entry. Under the root "synset" we find both internal relations ("synset relations") and ILI Relations, which link to ILI synsets.

Figure 5.3 shows the role played by the ILI as set of pivot nodes

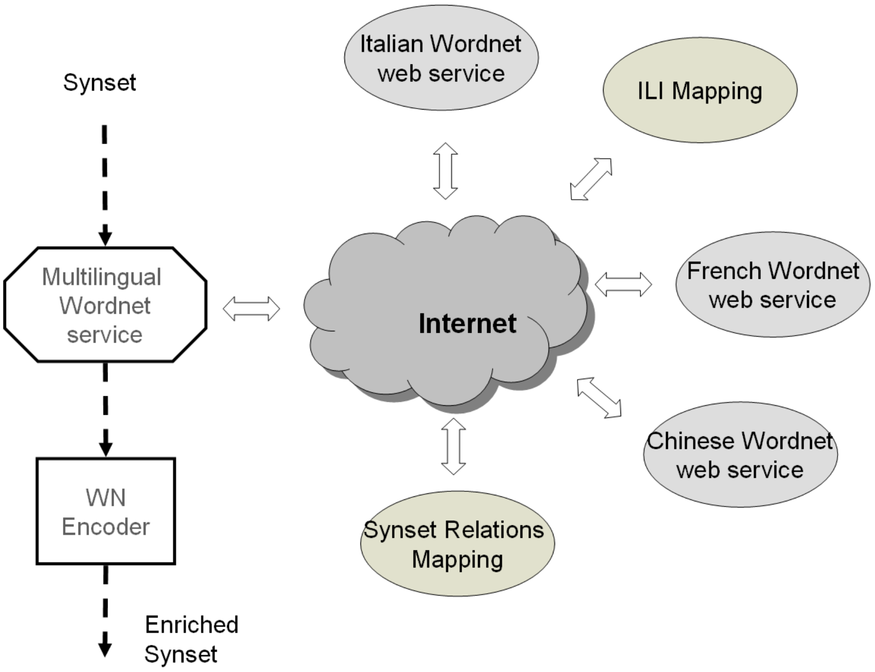

Synset
Multilingual Wordnet service
WN Encoder
Enriched Synset
Italian Wordnet web service
ILI Mapping
Internet
French Wordnet web service
Chinese Wordnet web service
Synset Relations Mapping

Figure 5.1: Multilingual Wordnet Service Architecture

allowing the linkage between concepts belonging to different wordnets.

In the Multilingual WN Service, only equivalence relations of type EQ_SYNONYM and EQ_NEAR_SYNONYM have been taken into account, being them the ones used to represent a translation of concepts and also because they are the most exploited (for example, in IWN, they cover about the 60% of the encoded equivalence relations). The EQ_SYNONYM relation is used to realize the one-to-one mapping between the language-specific synset and the ILI, while multiple EQ_NEAR_SYNONYM relations (because of their nature) might be encoded to link a single language-specific synset to more than one ILI record. In Figure 5.4 we represented the possible relevant combinations of equivalence relations that can realize the mapping between synsets belonging to two languages. In all

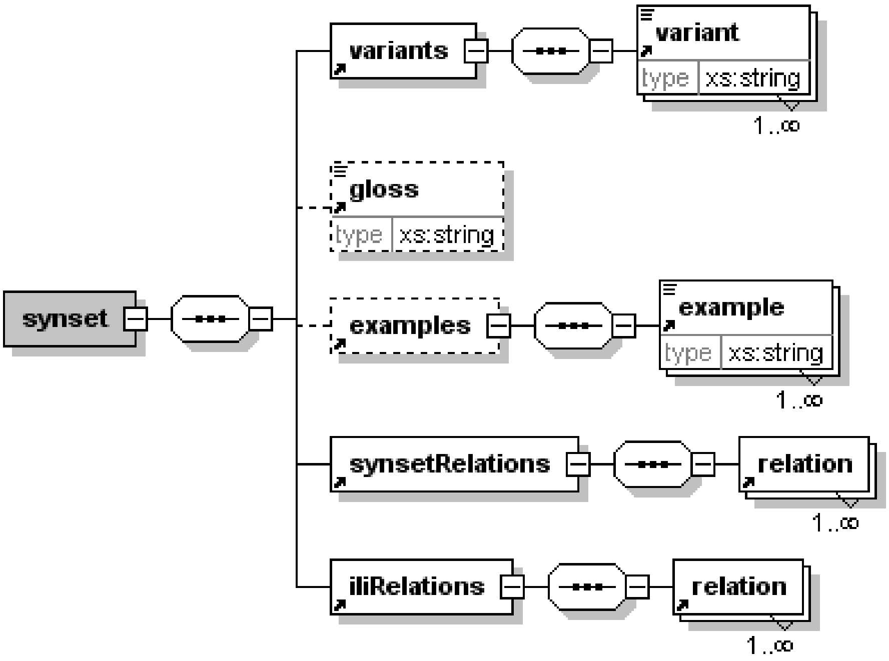


Figure 5.2: Schema of Wordnet Synsets Returned by WN Web Services

the four cases, a synset "a" is linked via the ILI record to a synset "b" but a specific procedure has been foreseen in order to calculate different "plausibility scores" to each situation. The procedure relies on different rates assigned to the two equivalence relations (rate "1" to EQ_NEAR_SYNONYM relation and rate "0" to the EQ_SYNONYM). In this way we can distinguish the four cases by assigning respectively a weight of "0", "1", "1" and "2".

The ILI is a quite powerful yet simple method to link concepts across the many lexicons belonging to the WordNet-family. Unfortunately, no version of the ILI can be considered a standard and often the various lexicons exploit different version of WordNet as ILI . This is a problem that is handled at web-service level, by in-

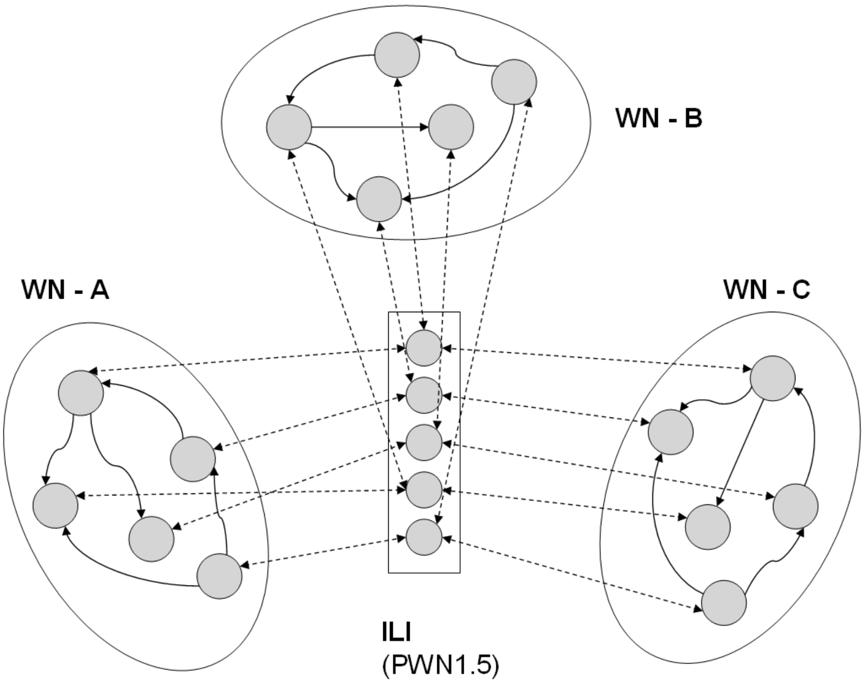


Figure 5.3: Interlingual Linking of Language-specific Synsets

corporating the conversion tables provided by [DPR]. In this way, the use of different versions of WN does not have to be taken into consideration by the user who accesses the system but it is something that is resolved by the system itself . This is why the version of the ILI is a parameter of the query to web service.

5.3 Description of the Procedure

On the basis of ILI linking, a synset can be enriched by importing the relations contained in the corresponding synsets belonging to another wordnet. In the procedure adopted, the enrichment is performed on a synset-by-synset basis. In other words, a certain synset is selected from a wordnet resource, say WN(A). The cross-lingual

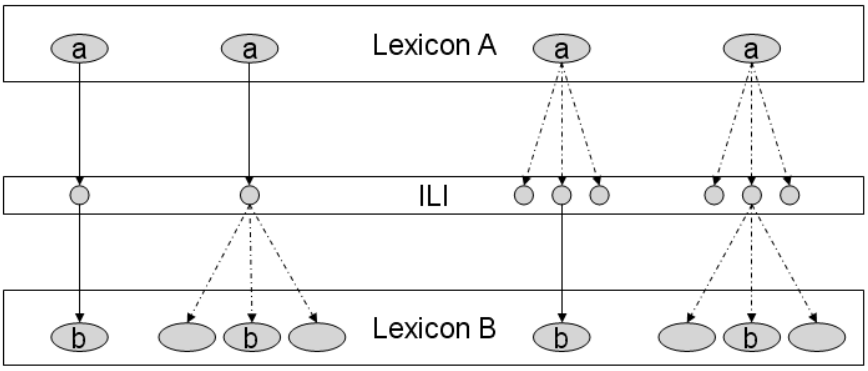


Figure 5.4: Possible Combinations of Relations between two Lexicons A and B and the ILI

module identifies the corresponding ILI synset, on the basis of the information encoded in the synset. It then sends a query to the WN(B) web service providing the ID of ILI synset together with the ILI version of the starting WN. The WN(B) web service returns the synset(s) corresponding to the WN(A) synset, together with reliability scores. If WN(B) is based on a different ILI version, it can carry out the mapping between ILI versions (for instance by querying the ILI mapping web service). The cross-lingual module then analyzes the synset relations encoded in the WN(B) synset and for each of them creates a new synset relation for the WN(A) synset. If the queried wordnets do not use the same set of synset relations, the module must take care of the mapping between different relation sets. In our case-study no mapping was needed, since the two sets were completely equivalent. Each new relation is obtained by substituting the target WN(B) synset with the corresponding synset WN(A), which again is found by querying back the WN(A) web service (all these steps through the ILI). The procedure is formally defined by the formula 5.5:

Every local wordnet has to provide a web service API with the following methods:

```
Let $a_j \epsilon$ A
Let $Ba_j$={$b_i$ | $b_i \epsilon$B and ($b_i$ ILI $a_j$)}
$\forall$ $b_i \epsilon Ba_j$
   Let $R_i$={$b_i r_k b_p$ | $b_i$,$b_p \epsilon$B and ($r_k$ $\epsilon$ $R_A \cap R_B$)}
   $\forall$ $b_i r_k b_p$ $\epsilon$ $R_i$
      Let $Ab_p$={$a_i$ | $a_i$ $\epsilon$ A and ($a_i$ ILI $b_p$)}
      $\forall$ $a_t$ $\epsilon$ $Ab_p$
         $a_j r_k a_t$ is a candidate relation
```

Legenda:

A,B	lexicons
a_j,b_i	synsets
$a_j r_p a_i$	synset relation r_p between a_j and a_i
b_iILIa_j	b_i is connected by ILI with a_j
R_A,R_B	relation space of lexicons B
$R_A \cap R_B$	the common relation space of B and A

Figure 5.5: Procedure for inferring new semantic relations

1. GetWeightedSynsetsByIli(ILIid, ILIversion)
2. GetSynsetById(sysnsetID)
3. GetSynsetsByLemma(lemma)

The returned synsets of each method must be formatted in XML following the schema depicted in Figure 5.2: The scores returned by the method "GetWeightedSynsetsByIli" are used by our module to calculate the reliability rating for each new proposed relation.

5.3.1 A Case Study: Cross-fertilization between Italian and Chinese Wordnets.

We explore this idea with a case-study involving the ItalianWordNet [RA03] and the Academia Sinica Bilingual Ontological Wordnet [CRHL04]. The BOW integrates three resources: WordNet, English-Chinese Translation Equivalents Database (ECTED), and SUMO (Suggested Upper Merged Ontology). With the integration of these three key resources, Sinica BOW functions both as

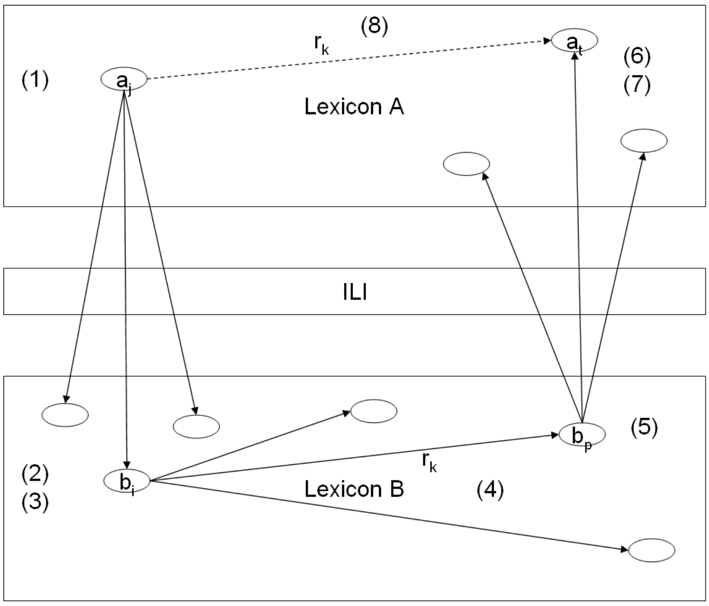


Figure 5.6: Finding New Relations

an English-Chinese bilingual wordnet and a bilingual lexical access to SUMO. Sinica Bow currently has two bilingual versions, corresponding to WordNet 1.6. and 1.7. Based on these bootstrapped versions, a Chinese Wordnet [CRHC05] is under construction with handcrafted senses and lexical semantic relations. For the current experiment, we have used the version linking to WordNet 1.6. ItalWordNet was realized as an extension of the Italian component of EuroWordNet. It comprises a general component consisting of about 50,000 synsets and terminological wordnets linked to the generic wordnet by means of a specific set of relations. Each synset of ItalWordNet is linked to the Interlingual-Index (ILI). The two lexicons refer to different versions of the ILI (1.5 for IWN and 1.6 for BOW), thus making it necessary to provide a mapping between the two versions. On the other hand, no mapping is necessary for the set of synset relations used, since both of them adopt the same set. For the purposes of evaluating the cross-lingual module, we have developed two web-services for managing a subset of the two resources. The Figure 5.7 shows a very simple example where our procedure discovers and proposes a new meronymy relation for the Italian synset passaggio,strada,via. This synset is equivalent to the ILI “road, route” that is ILI-connected with chinese synset (dao_lu, dao, lu) (Figure 5.7, A) . The Chinese synset has a meronymy relation with the synset (wan) (B). This last synset is equivalent to the ILI “bend, crook, turn” that is ILI-connected with Italian WordNet synset “curvatura, svolta, curva” (C). Therefore the procedure will propose a new candidate meronymy relation between the two Italian WordNet synsets (D).

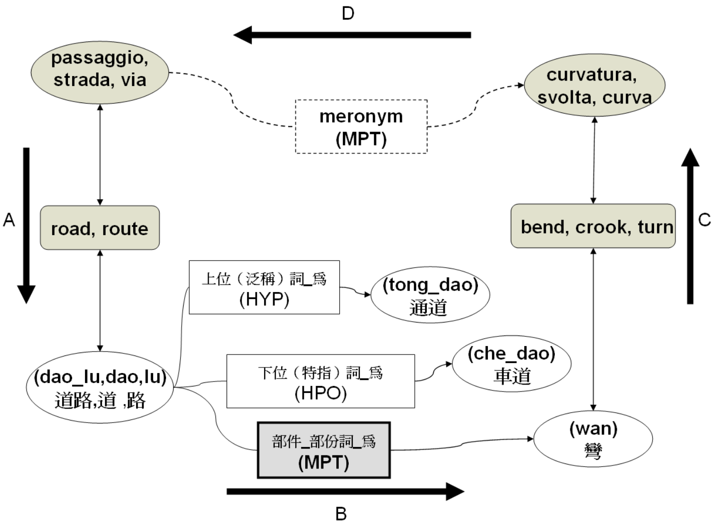


Figure 5.7: Example of a New Proposed Meronymy Relation for Italian

Similarly, Figure 5.8 shows the flow of information between the two WordNets.

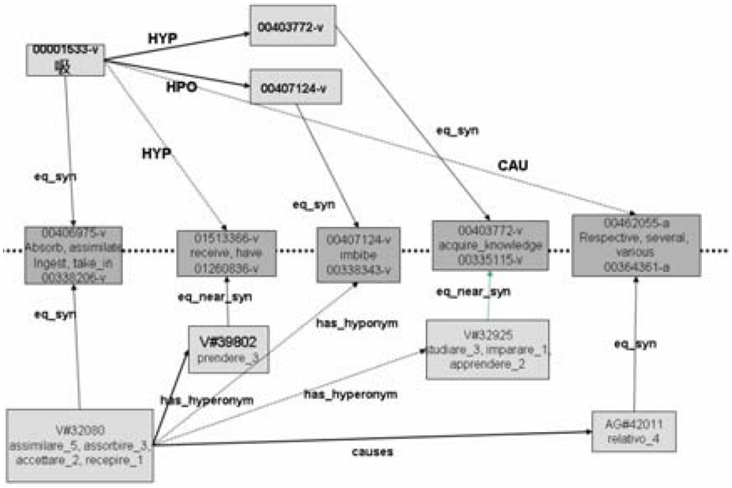


Figure 5.8: Inferred relations for Italian and Chinese.

5.4 Considerations and Lessons Learned

Given the diversity of the languages for which wordnets exist, we note that it is difficult to implement an operational standard across all typologically different languages. Work on enriching and merging multilingual resources presupposes that the resources involved are all encoded with the same standard. However, even with the best efforts of the NLP community, there are only a small number of semantic resources encoded in any given standard. In the current work, we presuppose a de-facto standard, i.e. a shared and conventionalized architecture, the WordNet one. Since the WordNet framework is both conventionalized and widely followed, our system is able to rely on it without resorting to a more substantial and comprehensive standard. In the case, for instance, of integration of lexicons with different underlying linguistic models, the availability of the MILE [NC03] was an essential prerequisite of our work. Nevertheless, even from the perspective of the same model, a certain degree of standardization is required, at least at the format level. From a more general point of view, and even from the perspective of a limited experiment such as the one described in this paper, we must note that the realization of the new vision of distributed and interoperable semantic resources is strictly intertwined with at least two prerequisites. On the one side, the semantic resources need to be available over the web; on the other, the language resource and semantic web community will have to reconsider current distribution policies, and to investigate the possibility of developing an "Open Source" concept for semantic resources.

Our proposal to make distributed wordnets interoperable has the following applications in processing of lexical resources:

1. Enriching existing resources: information is often not complete in any given wordnet: by making two wordnets interoperable, we can bootstrap semantic relations and other information from other wordnets.

2. Creation of new resources: multilingual lexicons can be bootstrapped by linking different language wordnets through ILI.

3. Validation of existing resources: semantic relation information and other synset assignments can be validated when it is reinforced by data from a different wordnet.

In particular, our work can be proposed as a prototype of a web application that would support the Global WordNet Grid initiative[1]. Any multilingual process, such as cross-lingual information retrieval, must involve both resources and tools in a specific language and language pairs. For instance, a multilingual query given in Italian but intended for querying English, Chinese, French, German, and Russian texts, can be send to five different nodes on the Grid for query expansion, as well as performing the query itself. In this way, language specific query techniques can be applied in parallel to achieve best results that can be integrated in the future. As multilingualism clearly becomes one of the major challenges of the future of web-based knowledge engineering, WordNet emerges as one leading candidate for a shared platform for representing a

[1] http://www.globalwordnet.org/

lexical knowledge model for different languages of the world. This is true even if it has to be recognized that the wordnet model is lacking in some important semantic information (like, for instance, a way to represent the semantic predicate). However, such knowledge and resources are distributed. In order to create a shared multilingual knowledge base for cross-lingual processing based on these distributed resources, an initiative to create a grid-like structure has been recently proposed and promoted by the Global WordNet Association, but until now has remained a wishful thinking. The success of this initiative will depend on whether there will be tools to access and manipulate the rich internal semantic structure of distributed multi-lingual WordNets. We believe that our work on LeXFlow offers such a tool to provide interoperable web-services to access distributed multilingual WordNets on the grid. This allows us to exploit in a cross-lingual framework the wealth of monolingual lexical information built in the last decade.

Part III

Exploiting Semantic Resources

Chapter 6

Semantic Folksonomies

Abstract

Collaborative tagging is a new content sharing and organizational trend and refers to the process by which many users add metadata in the form of keywords to shared content. By analyzing the current structure and usage patterns of collaborative tagging systems we discovered many important aspects which still need to be improved in order to bring tagging systems to their full potential. Examining the main causes of decrease in precision and recall in tag-space based searches (synonymy, polysemy, different lexical forms) we can infer that most of them may be solved adding semantics to collaborative tagging systems. In this Chapter we propose a model of semantic collaborative tagging and we analyze how semantic resources can be exploited in this context. When a user decides to tag resources, he must be able to disambiguate each tag, defining its semantics. Moreover, we introduce properties to link concepts to a specific resource. This process will be referred to as *semantic collaborative tagging.* In this way, the outcome of semantic tagging activity consists of *producing a set of unambiguous assertions on resources* Each of them could represent statements about the topic or kind of resource or concern the user opinion about the web resources.

6.1 Collaborative tagging systems

During the last few years, the Web has experienced the growing diffusion of many kinds of collaborative tagging systems and the related increase of communities of taggers [CM06] [Wei05]; they are actively involved in the process of labelling and cataloguing resources of interest, exploiting the growing amount of information collected to improve their searches and content discovery process. Some of the most used and representative collaborative tagging services are [THS05]:

- **Del.icio.us** (http://del.icio.us): it allows users to assign a free set of tags to a Web resource identified by its URL; this kind of tagging schema is also known as 'social bookmarking', because users can create and share resource annotations in a way similar to local bookmarking systems integrated in existing browsers;

- **Flickr** (http://www.flickr.com): this is a photo sharing system; each user can share and tag his personal photos and access and tag photos of other users;

- **Technorati** (http://www.technorati.com): it allows authors to tag their blog posts, aggregating information contained in weblogs and facilitating their search.

All tagging systems listed above are usually adopted by particular communities of users; del.icio.us by Computer Science experts, Flickr mainly by amateur photographers and Technorati by bloggers.

As we can argue, also by reading this short description of significant examples, tagging represents a collaborative social effort of a community of users constituted around a tagging service; with his tagging action, every user, mainly on the basis of his interests, directly contributes to the creation of a shared metadata collection, progressively augmenting the relevance and the richness of shared data. The three main components of collaborative tagging systems are: users, resources and tags [LA05]. Users may be grouped according to their common interests; resources may be related by the different kinds of links which constitute the basis of current Web; tags provide the connection between a single user and a particular resource. When every user can assign a freely defined set of tags to a resource, *the tag collection will reflect the social attitudes of the community of users* and a shared social organization and structuring of the tag-space will emerge: this phenomenon is referred to as emergent semantics. It continuously adapts the tag space to the way users choose to describe resources, reflecting their tagging behavior.

The result of this process of adaptive social structuring of the tag-space in a collaborative tagging system has recently been defined as folksonomy [Smi04]. A folksonomy is a combination of two words: 'folk' and 'taxonomy'. 'Folk' is used to indicate the social collaborative component of the process of tags definition; 'taxonomy' instead refers to the method of organizing concepts in predefined and sometimes rigid structures, in order to better define their semantics and relations. When we speak about folksonomy, we refer to the collaborative and progressive definition of a relaxed categorization and organization of content, not based on a rigid hierarchical structure, and the related emergent semantic specification of concepts, i.e. of the meaning of tags. In this way the user has the freedom to choose autonomously his tags.

Many formal (research articles references [Bec06]) and informal (blogs references [She05] [Kro]) analysis of collaborative tagging system have also identified the low user learning curve and the relatively little bootstrapping cost of this kind of services as two relevant factors influencing their spread and rapid diffusion.

Analyzing in more depth the current structure and usage patterns of collaborative tagging systems, we can discover many important aspects which still need to be improved so as to really exploit their real potential.

6.2 Weak points of current collaborative tagging systems

When we analyse existing collaborative tagging systems, we can point out some relevant weak features; in particular, many of them can be related to the insufficient semantic information in the process of assigning descriptive keywords to a resource ([GH05], [ZXS06], [CM06], [GT06], [Mat04]). As a consequence, we can identify the following main causes of weakness:

- **Polysemy** (6.2.1)
- **Synonymy** (6.2.2)
- **Different lexical forms** (6.2.3)
- **Misspelling errors or alternate spellings** (6.2.4)
- **Different levels of precision** (6.2.5)
- **Different kinds of tag-to-resource association** (6.2.6)

Through the following example, we give a summary of the most important weak points of existing collaborative tagging systems listed above. Let suppose that there are four different Web users: John, Monica, Bill and Anne. John, Monica and Bill are browsing the same Web resource speaking about a new model of Jaguar, a British luxury car manufacturer and decide to tag it. They have in mind to state that the Web resource is about cars, intended as the concept of four wheels vehicle. Anne is also browsing and is searching information about the jaguar, the large spotted feline; after a lot of Web searching activity, she decides to tag the jaguar section of an interesting web site about jaguar felines. All those situations are represented in Figure 6.1. Every user may freely choose one or more tags (character strings) to describe a Web resource. John, Monica and Bill refer to the concept of car choosing only one tag; John uses the word ‘automobile’ (one of the several synonyms associated to the concept of car), Monica the plural form of the word

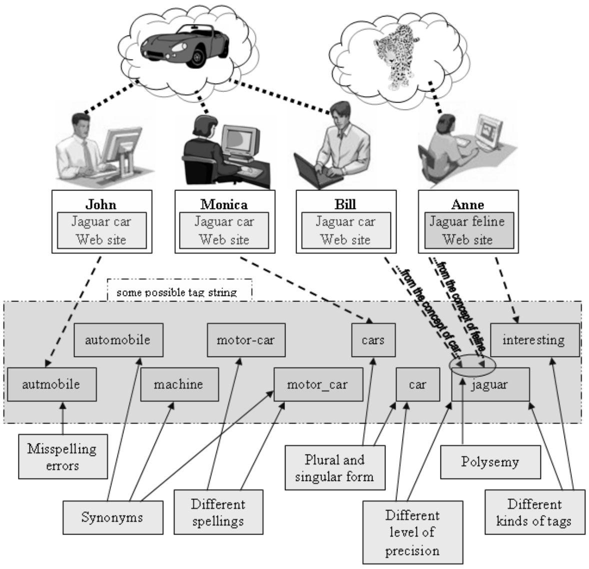


Figure 6.1: The tag choice problems

'car' (different lexical form), Bill better specifies the concept with an increased level of precision using the word 'jaguar' (different level of precision) that has also another meaning, the one intended by Anne who adopts the same tag to refer to the large spotted feline (polysemy). Moreover, 'motor-car' and 'motor_car' are possible different spelling of the same word and 'autmobile' represents a possible user spelling error during the tagging of a resource. Finally Anne links two tags to the visited Web resource, 'jaguar' and 'interesting' with a different purpose: the first one to describe the topic of the resource and the second one to express his personal opinion about the resource (different kinds of tag-to-resource association).

We have considered two main parameters to evaluate tag based searches in terms of their retrieval effectiveness. These parameters are usually adopted to measure how well an information-retrieval system, in this case a tag based search system, is able to execute a specific search [SS02]:

- **Precision** : the percentage of all retrieved resources that are actually relevant to the query;

- **Recall** : the percentage of all relevant resources present in the system that are returned by the search.

Generalizing, most problems of current collaborative tagging systems can be traced back to the existence of $n : m$ relations between concepts and tags used to identify an intended concept. When a single tag is used to express different concepts ($Tag(1) \longrightarrow Concept(n)$), polysemy issue occurs. When we adopt that tag to find all resources related to a specific intended concept, *precision decreases* because of the noise generated by the other retrieved resources dealing with different concepts but identified by the same tag. In Figure 6.2 we show an example of result noise, depicted as gray area, generated when we search for all the resources tagged with the keyword 'machine' meaning the concept of four wheels vehicle.

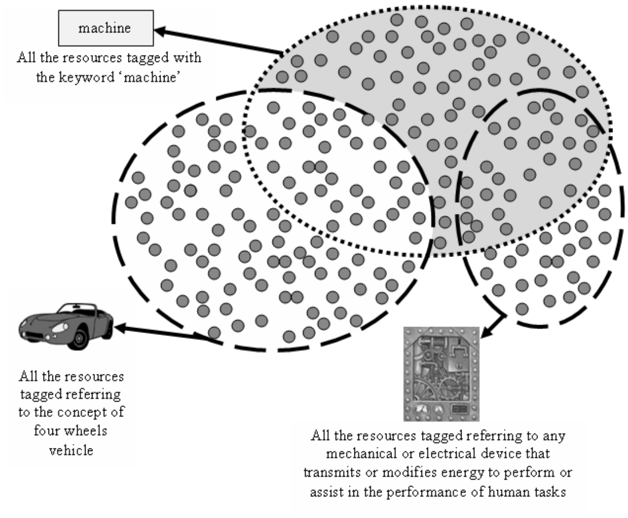

machine
All the resources tagged with the keyword 'machine'
All the resources tagged referring to the concept of four wheels vehicle
All the resources tagged referring to any mechanical or electrical device that transmits or modifies energy to perform or assist in the performance of human tasks

Figure 6.2: A single tag used to express different concepts

On the other side, multiple tags can be used to refer to the same concept ($Tag(n) \longrightarrow Concept(1)$); this occurs in case of synonymy, different lexical forms, misspelling errors or alternative spellings. In this case, using one of the different tags to refer to a concept and find all related resources, *recall decreases* because of the presence of other relevant resources related to the same concept that are not retrieved since they are tagged using distinct words. In Figure 6.3 we show an example of recall lowering that occurs when we search for all the resources tagged with the keyword 'machine', but we are not able to retrieve all the resources tagged thinking about the concept of four wheels vehicle using different tags ('cars', 'auto', 'automobile' and 'car').

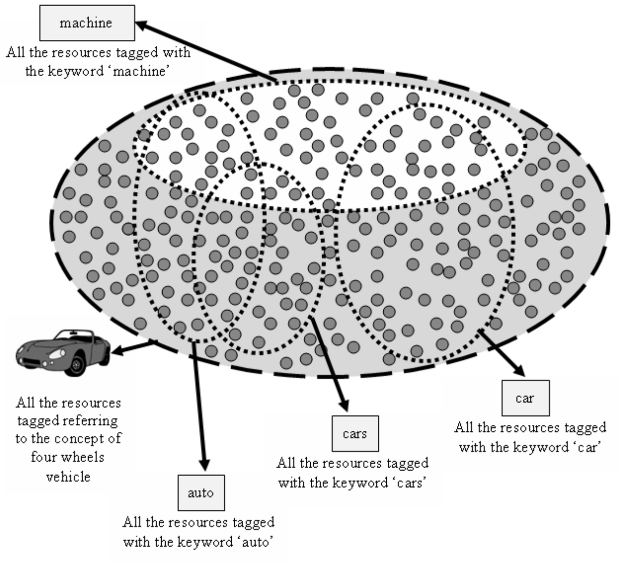

machine
All the resources tagged with the keyword 'machine'
All the resources tagged referring to the concept of four wheels vehicle
auto
All the resources tagged with the keyword 'auto'
cars
All the resources tagged with the keyword 'cars'
car
All the resources tagged with the keyword 'car'

Figure 6.3: The same concept referred by different tag

6.2.1 Polysemy

When a user performs a tag-based search, he needs to properly modify the search tag set to increase precision and recall. Usually a set of tags is used to specify a particular meaning; in fact we can notice that, during a tag-based search, one or more new tags are often added to the search tag set to disambiguate the meaning of tags already present.

If the user of a social tagging system wants to find information about the Jaguar car manufacturer, he could type only the word 'jaguar' to form the search tag set, obtaining as result every Web resource tagged with this word. Obviously only part of this result is of real interest to the user, in fact the word 'jaguar' may have a lot of other meanings (polysemy [GH05], [CM06]): a large felid animal, the codename of an Apple operating system, a video game console made by Atari, a guitar built by Fender, etc. In such a situation the user usually adds other tags to the search tag set, in order to better define the intended meaning; for instance, he could choose the tag 'car' to find all resources tagged with 'jaguar' and 'car', as an attempt to increase the precision of his search. Considering del.icio.us, the most popular tags used to refer to the 'automobile' word are: 'car', 'cars', 'auto' and 'automotive'. As a consequence we can assume that when a user wants to specify the meaning of the tag 'jaguar', he will usually add one of the four tags just mentioned.

When we search for all Web resources tagged using 'jaguar', we obtain 1450 results[1]. In Table 6.1 we can see the four subsets of this resources group produced adding to the tag 'jaguar' one of the most popular tags used to refer to the 'automobile' word, mentioned before and called disambiguation tags.

Search tag set	**Number of Web resources found**
jaguar car	217
jaguar cars	183
jaguar auto	75
jaguar automotive	37

Table 6.1: Single disambiguation tag

[1] All the numeric data reported in the examples included in this chapter are obtained querying del.icio.us (http://del.icio.us) in date 3.1.2007.

In this way, we obtain a first refinement of the search results and an increase in precision. Now we better analyze the structure of these four sets of resources to understand their level of overlapping and the size of their intersections. To do this, we examine the number of Web resources tagged with all possible combinations of the four disambiguation tags ‘car’, ‘cars’, ‘auto’ and ‘automotive’.

In what follows, we suppose that the total number of resources relevant to our search (all resources dealing with Jaguar cars) is equal to the number of different resources identified by the four search results shown in Table 6.1. Thus, we can determine the supposed total number of relevant resources present in the tagging system applying the ‘Inclusion-exclusion principle’ [Mat]. It is used to compute the cardinality of a set composed by the union of other finite sets, through the cardinality of their intersections. Supposing to have n finite sets A_1, A_2, A_3, $\ldots$, A_n, we can compute the cardinality of their union using the following formula:

$$\sum_{i=1}^{n}((-1)^{i+1}\sum_{0\leq j_1\leq ..\leq j_n}\bigcap_{k=1}^{i}(A_{j_k}))$$

Applying the ‘Inclusion-exclusion principle’ to compute the total number of relevant resources we obtain the result of 324.

Starting from the previous analysis of the tag space, we can notice that when we use only one disambiguation tag among the most popular tags used to refer to the automobile word, we obtain the results represented in Table 6.2 in terms of recall.

Only a relatively small part of all relevant resources present in the system is selected and shown as search result to the user; this fraction ranges between 11% and 67%, depending on the popularity of the disambiguation tag added to the tag ‘jaguar’ to form the search tag set.

6.2.2 Synonymy

Besides polysemy, another search limitation in the existing tagging systems is due to synonymy ([GH05], [Mat04]) that is the presence of different tags/words having the same meaning. For example,

Search tag set	**Recall**
jaguar car	217/324 = 0.6697 - **67%**
jaguar cars	183/324 = 0.5648 - **56%**
jaguar auto	75/324 = 0.2315 - **23%**
jaguar automotive	37/324 = 0.1142 - **11%**

Table 6.2: Recall with one disambiguation tag

we can refer to a computer using tags like 'computer' or 'pc' or to automobiles using tags like 'car', 'auto', 'automobile', etc. When we search all Web resources dealing with computer, we choose a tag so as to identify this concept, excluding from the result all the relevant resources tagged using its synonyms; as a consequence we must face the following situation graphically represented by a Venn diagram in Figure 6.4.

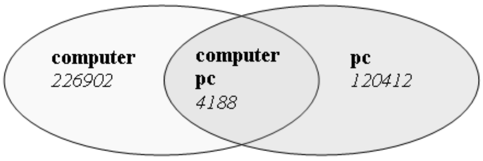


Figure 6.4: The relevant Web resource partition due to different synonyms of the word computer.

Only 4188 Web resources, compared to a total number equal to 343126 (1.2%), have been tagged using both tags, maybe by expert users in order to relieve the search limitations caused by synonymy. But we can't rely on the users' tagging behaviour, expecting that the user will add all possible synonyms of every word used when he tags a resource; it could also reduce the handiness of tagging.

Often *the limitations caused by synonymy are strictly related to those caused by polysemy.* In fact, it is possible that a word or tag has more than one meaning, but it presents also many synonyms. In such a situation the search is complicated by the presence of both polysemy and synonymy. Due to polysemy, the user will need to add other tags in order to disambiguate those already chosen together with all problems related in terms of recall and precision. Furthermore, all resources tagged with tags synonym to the one chosen and therefore potentially relevant, will not be included among the search results.

6.2.3 Different lexical forms

Problems similar to those previously described may often arise as a consequence of using different lexical forms ([GT06]) to refer to the same concept.

Plural nouns, different verb conjugation and name-adjective couples.

Referring to the previous example about polysemy, the tags 'car' and 'cars', both used to indicate the concept of automobile, are the *singular and the plural form* of the same word, but are managed as different entities. In fact, if we consider a search tag set composed by the tags 'jaguar' and 'car', the system will return 217 resources, omitting those tagged with 'jaguar' and 'cars' but without the tag 'car' and thus preventing the user to access to other 65 relevant Web resources.

We can observe those problems also in the e-commerce Web sites tagging. E-commerce web sites are often tagged with 'buy' or 'buying' (*the gerundive form of buy*). When the user wants to find all the Web sites that sell scooters, depending on the search tag set used, 'scooter buy' or 'scooter buying', he will retrieve different sets of results (without considering the problems caused by the presence of polysemy and synonymy regarding the tags used).

A similar case of search precision loss happens when a document describing different kinds of sources of energy is tagged by different users respectively with the keywords ‘energy’ and ‘energetic’; both tags express very similar meaning but *the former is a noun and the latter the respective adjective*. When a user asks for all the resources speaking about ‘energy’, only the first set of the two ones will be showed as search result (obviously including their intersection). The same problem may arise with couples of very similar keywords like: ‘pollute’ - ‘pollution’, ‘dance’ - ‘dancing’, etc.

Multi-word tags.

In many tagging system there are a lot of tags composed by more than a single word: *multi-word tags* (‘semantic web’, ‘personal computer’, ‘web design’, etc.). When a user tags a Web resource, if he divides the words that constitute the tag using blank characters, the system will consider all those words as different tags and not as a lexical form referring to a single concept.

Moreover, to overcome this problem, the users of tagging systems usually adopt different solutions and thus different lexical forms to refer to the same concept, causing a search space partitioning. For instance, we suppose to retrieve all resources speaking about the Semantic Web. Some of these are tagged with two separate tags: ‘semantic’ and ‘web’. They will be included in every search even if only one of these two tags constitutes the search tag set. Other alternative tags which refer to the same concept are: ‘semWeb’, ‘semanticWeb’, etc. It is also possible that a single user or a community of users defines a new tag to refer to the Semantic Web, for example ‘sWeb’. When we search for Semantic Web related resources, if we type the tag ‘semWeb’, we will identify 5387 resources, missing other 9435 results tagged with ‘semanticWeb’ and not with ‘semWeb’ (represented in the Venn diagram shown in Figure 6.5).

One aspect of current tagging systems, related to multi-words tags, is represented by *the different notations* which could be adopted by users. For instance, when a user chooses to collapse a tag composed by different words he could use the CamelCase notation (re-

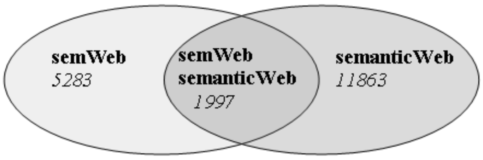


Figure 6.5: Web resource partition due to different multiword tags referring to the Semantic Web.

moving all blank spaces and starting every word besides the first one using an upper-case character) or he could replace the blank characters with other characters like underscore, slash, dot, etc. Also in this case their effect is a bad tag space partitioning.

6.2.4 Misspelling errors or alternate spellings

Also misspelling errors or alternate spellings ([GT06]) represent possible sources of search imprecision; indeed, when a user makes a mistake typing a tag, he isolates the selected resource decreasing the possibility of a future retrieval, especially if the considered resource isn't popular. He can *misspell a tag* related to a document which describes Condoleezza Rice using the keyword 'Condoleeza' (only one 'z'), isolating it from the others. Moreover, a user could write the word/tag 'colour' in a slightly different way, adopting *a different spelling*: 'color'. They both identify the same concept. As shown in Table 6.3 in del.icio.us there are 72949 web resources tagged with at least one tag among 'color' and 'colour'; only 2698 of them belong to their intersection and thus are tagged with 'color' and 'colour'.

Different forms of spelling could be present especially when we refer to *proper names*, *acronyms* or *word punctuation*, for example 'Al-Jazeera' and 'AlJazeera' are different tags which refer to the same concept. We could also include in this broad category those problems created by *different rules of capitalisation.*

Search tag set	**Number of Web resources found**
color	61446
colour	14101
color colour	2698

Table 6.3: Different spelling of the same word

6.2.5 Different levels of precision

Another problem which may occur during the tagging process is related to the different level of precision that could be adopted by choosing a keyword to describe and characterize a resource; this question is also referred as “the basic level” problem ([GH05]). Let suppose that a user wants to tag a Web page about a new musical record review. Depending on the different level of experience and musical knowledge and on the aim and the accuracy of his tagging behaviour, he could choose a general keyword like ‘music’ or a more specific one, e.g. ‘jazz’. A tag-space based search regarding all resources tagged with the tag ‘music’ will not find those tagged with ‘jazz’, lowering the recall. Similarly a user could tag a document which describes Java programming language with ‘programming’ (general) or ‘Java’ (more specific). Different users could be characterized by different levels of precision and every user generally adopts a personal level of precision while tagging; the level of specificity of keywords is influenced by the aim of tagging but also by the knowledge and the expertise of the user.

6.2.6 Different kinds of tag-to-resource association

Analysing the keywords used in existing tagging systems and the tagging behaviour of users, we can define different implicit kinds of relations that links a tag to a specific resource. Indeed the association of a tag to a resource is made without specifying the relation. Other times the tag represent the topic of the resource, other times it is a sort of rating about the resource or a simple and often personal memo information.

Through a tag analysis, it is possible to group tags in distinct sets on the basis of the kind of relation which associates the tag to a particular resource. Several research papers have proposed possible categorizations of the kinds of tags. In [GH05] seven different groups of tags are identified by analysing del.icio.us; in [CM06] those seven sets are grouped together in three main categories: tags which identifies properties of the resource referred, tags which describe the contents of a resource in terms of its relation to the tagger and tags which collect information about a particular task in which the tagger is involved. Also [ZXS06] proposes five groups of tag types: *content based tags* ('computer', 'AMD', 'programming', etc.); *context based tags* (for example those tags which describe location and time to specify the context in which the resource was created or saved, e.g. 'Rome', '10-10-2001'); *attribute tags* (which describe a resource but could not be directly derived from its content, e.g. 'Jimmy's blog', 'post' etc.); *subjective tags* ('interesting', 'funny', etc.); *organizational tags* ('mywork', 'toread', etc.).

Another relevant problem briefly mentioned before is represented by *the presence of tags, or better of lexical forms expressed in different languages.* A tagging system is used by communities of people from different countries and even if English is the mainly adopted language over the Web, the multilanguage support is an important issue in a global Web system. From the analysis made in [GT06], we observe that about 64% of the set of tags present in del.icio.us are valid English language dictionary words and about 32% of the same set is unrelated to a particular language (proper names, acronyms, etc.). Therefore, more than 95% of the total tag set of del.icio.us is expressed using the English language or at least consistent to it. Besides the English, on del.icio.us other used languages are Spanish, German, French and Portuguese.

6.3 Semantic collaborative tagging

Examining the different causes of inconstencies and loss of precision in tag-space based searches, we can infer that most of them may be solved or substantially reduced bringing semantics to collaborative tagging systems. Each tag should not represent just a simple sequence of characters, but should be defined by specifying its meaning. When a user decides to tag resources, describing by means of one or more keywords, he must be able to disambiguate each of them, defining their semantics or better pointing out their contextualized meaning. Moreover, we introduce properties to link concepts to a specific resource; this process will be referred to as *semantic collaborative tagging.* In this way, the outcome of semantic tagging activity consists of *producing a set of unambiguous assertions on resources*: **semantic assertions**. Each of them could represent statements about the topic or kind of resource or concern the user opinion about the resources.

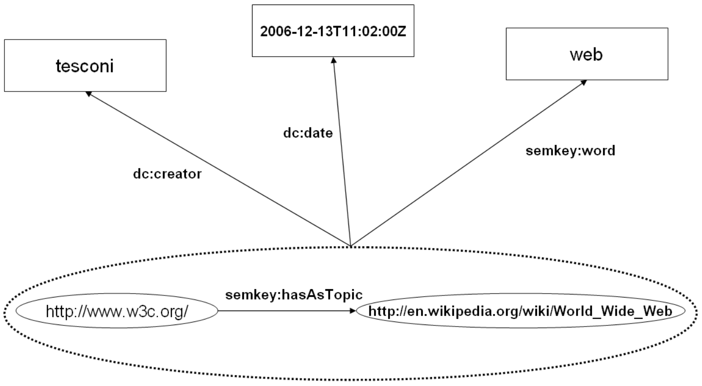


Figure 6.6: Example of RDF triple

These assertions represent the classical RDF triples that are composed by the following parts:

- **Subject:** the URL of the Web resource.

- **Property:** the URI that identifies the relationships between a resource and a concept.

- **Object:** the URI that identifies the concept associated to the web resource.

Other data need to be added to those just mentioned in order to fully describe the association of a concept to a particular Web resource:

- the *lexical form* (string) employed by the user to identify the particular concept referred to at the time of generation of the semantic tag;

- the *username* adopted in our system in order to uniquely identify the user, author of the semantic annotation;

- the *date* and the *time* of generation of the semantic annotation.

These data are all descriptive information that is added to the core RDF-triple previously mentioned. To represent it we need to exploit another RDF expressive conventionalism: the reification [Fut06]. It is used to make RDF statements that describe an entire RDF triple; to do this we need to univocally refer to the RDF-triple that must be described assigning it an identifier. It could consist of an URI, which is unambiguous over the Web or of a blank node identifier which is unambiguous inside the local RDF document that contains it. This ID is formally assigned to the RDF-triple using other four additional RDF-triples in order to respectively specify its subject, its predicate, its object (*rdfs:subject*, *rdfs:predicate*, *rdfs:object*) and the class of belonging (*rdfs:statement*). Once this ID is determined, we can define other properties referred to the entire RDF-triple, in particular:

- **semkey:word** : the lexical form used to refer to the concept during the semantic tagging process;

- **dc:date** : date and time of generation of the semantic annotation;
- **dc:creator** : username of the user who generated semantic annotated data.

In the properties just described, the namespace ‘semkey’ refers to the local RDF Schema namespace of our semantic tagging reference and the namespace ‘dc’ refers to the Dublin Core Metadata RDF Schema namespace [Dub]. As a consequence the set of information related to the association of a disambiguated tag to a Web resource, made by a particular user in a precise time, is represented as showed in Figure 6.6. The main RDF-triple is represented by the information contained in the dotted circle; through the RDF reification conventionalism three other descriptive data are added to the main RDF-triple. If we want to represent those data using the XML/RDF serialization, we obtain the following schema:

```
<?xml version=’1.0’?>
<rdf:RDF xmlns:rdf=’http://www.w3.org/1999/02/22-rdf-syntax-ns#’
xmlns:semkey=’http://www.semkey.org/schema/’
xmlns:dc=’http://purl.org/dc/elements/1.1/’>

<!-- Triple 1 -->
<rdf:Description rdf:about=’http://www.w3.org/’>
<semkey:hasAsTopic rdf:nodeID=’id00001’
rdf:resource=’http://it.wikipedia.org/wiki/World_Wide_Web’/>
</rdf:Description>

<!-- Descriptive information added to triple 1 exploiting its
reification -->
<rdf:Description rdf:nodeID=’id00001’>
<semkey:word>web</semkey:word>
<dc:date>2006-12-13T11:02:00Z</dc:date>
<dc:creator rdf:resource=’http://www.semkey.org/users/tesconi’/>
</rdf:Description>
```

```
</rdf:RDF>
```

Adding semantic information to tags, search efficiency and effectiveness will be considerably improved and new important information access and organization patterns will be exploitable. In order to make it possible, a sort of shared ontology should be available and concepts expressed should be easily and univocally referenced. In Figure 6.7 we graphically schematize a possible scenario in which a shared ontology is used to semantically tag three resources referring to specific concepts.

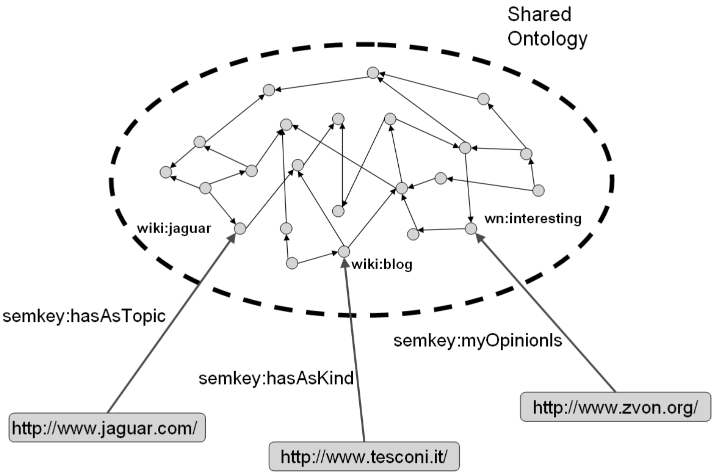


Figure 6.7: Example of semantic tagging: resources that reference a shared ontology

Such a kind of global generic-domain organization of concepts (the shared ontology) should be provided in a way which doesn't decrease the usability of the system and its adoption by a great number of users. At present, especially in specific and limited domain of interest (e.g. academic research, corporate knowledge management, medical classification, etc.), tagging systems are supported by *controlled vocabularies* [MM06a]; they consist of a set of terms structured and interconnected in order to specify their semantic relations. Using controlled vocabularies while tagging, we can manage more easily synonymy, polysemy, misspelling, plural words and other different lexical forms of a concept. They are adopted as a reference to define tags for a resource, mainly in a library or text cataloguing context. They are structured and kept up to date by experts of a specific domain. They also usually arrange their content in a hierarchical or taxonomical manner from more general to more specific concepts. Nevertheless controlled vocabularies present a rigid structure and are too strongly domain dependent to be a valid support to the definition of the semantics of a generic tagging system [Ros01]. Usually the language used by a community of taggers changes continuously, reflecting its social behaviour and it shouldn't be forced to adapt itself to the rigid constraints of structure of a controlled vocabulary or also to its limited set of terms. Moreover keeping up to date a controlled vocabulary is an expensive task because it involves domain experts and knowledge engineers; in this process few people define a structure of information used by many more users, which cannot directly take part to its definition [MH06].

Considering existing collaborative tagging systems, all these problems are absent because they leave the users complete freedom when choosing resource keywords. This results in *the social emergence of a defined structuring of the tag space, called folksonomy*, which continuously adapts itself to the way user communities tag resources and which is not limited by structural and organizational constraint. But when we leave such a freedom of tagging to the users all the problems and inconsistencies described in the previous section arise.

In order to define a semantic keyword disambiguating the meaning of a particular lexical form, we need to exploit some resources that should support the following tasks:

- starting from a particular lexical form it should identify all its possible meanings (or concepts), providing for example a short textual description for each one;

- it should allow for the setting of a univocal reference to each single concept.

Considering these fundamental requirements, we have identified two different and maybe complementary kinds of resource currently available over the Web:

- **WordNet**: a lexical database which is based on the concept of set of synonym words, called synset, which define a particular meaning; it is sufficiently structured and includes a lot of lexical and semantic relations between words and synsets. At present, WordNet version 3.0 [Wne] is available; it includes 117597 concepts (or distinct synsets).

  Wordnet [Wne] is updated by a group of lexicon experts and presents quite a complex net of internal relations, in fact it has been developed in order to support text mining and information extraction. WordNet has a broad coverage of all parts of speech (names, verbs, adverbs and adjectives).

- **Wikipedia**: the famous collaboratively-edited free encyclopedia, which represents the result of the efforts of many editors worldwide, directly involved in this project; it is rich of extensively described and easily referenced definitions of concepts and it is continuously increasing its dimension and completeness.

Wordnet could be used for disambiguating personal opinions of users (expressed by adjectives) about resources. Wikipedia does not cover all parts of speech like WordNet, but it is extremely rich and constantly updated. It provides descriptions of many specific *proper-named concepts* that are not present in WordNet and could be useful for creating topics assertions. Wikipedia is obviously less strongly structured than WordNet, but thanks to the possibility to collaboratively edit its data, it is *constantly enriched with new updated contents.* It supports the disambiguation of polysemous words through the introduction of *disambiguation pages* which allows users choosing a specific meaning among those available. When a word has several synonyms, Wikipedia uses the *redirect mechanism* to make them point to the same page. Moreover since May 2004 Wikipedia includes also a sort of relaxed classification system of its documents: the *Wikipedia categories*. Every description included in the encyclopedia can be assigned to one or more categories in order to provide a new way of accessing and cataloguing it. Users can create new categories arranging them in a hierarchical-like structure. Every document is also related to many other documents through simple links usually used to point to extended descriptions of terms. In Table 6.4 we show some important numerical data [Wikb] regarding the English version of Wikipedia in order to quantify the great amount of information collected. For more information see [Wika].

Number of articles included	1,4 Millions
Number of active editors (who edited at least 10 times since they arrived)	150.000
Number of links between Wikipedia articles	32,1 Millions
Number of redirects	1,4 Millions
Number of categories	176.000
Percentage of categorized articles	86%

Table 6.4: Wikipedia statistics - English - October, 2006

The exploitation of Wikipedia for building a collaboratively edited collection of concepts and relations between them is an important opportunity [DMW06] [Vos06] [MH06]. It is possible to extract from Wikipedia a relaxed controlled vocabulary or thesaurus-like structure which can be used to support the semantic tagging activity. The flexible and looser set of semantic connections that characterize thesauri, usually represented by equivalence, hierarchy and related-concepts relations, compared with rigid classification systems, can provide the right degree of structuring for such a collaboratively maintained resource and Wikipedia constitutes a considerable collection of data and relations that may be exploited to bootstrap it. It is also relevant, but may be initially too difficult to organize and maintain, the possibility to extend Wikipedia with an increased sets of user defined semantic relations; this is an attempt to fully import Semantic Web vision in this socially edited resource [MK06] [Pla].

At present, an interesting project that is attempting to build a sort of social ontology which may be useful to support the disambiguation of concepts and their unambiguous references during semantic tagging activity is represented by OmegaWiki [Ome]. It is a free, multilingual resource with lexicological, terminological and thesaurus information. It is substantially a collection of concepts; each of them is characterized by a short description and one or more strings that refers to it. All these data associated to a concept can be provided adopting different languages and some simple type of relation between concepts can be estabilished. OmegaWiki is still in an intial phase of development; it aims to be collaboratively-edited: it relies on social editing efforts as its fundamental growing factor. Currently users still have read-only access and only a group of testers have the possibility to modify OmegaWiki contents; anyway its collection of data is considerably growing.

6.4 Requirements and global architecture

In this section we describe the main architectural chooses faced when structuring the proposed social semantic tagging system: SemKey. First of all we identify and specify in more detail its requirements, its desired features. Then we examine the global architecture and the more relevant organizational issues describing and justifying the decisions made, but also mentioning ideas about possible future relevant improvements.

6.4.1 General requirements

The main idea that supports our system, which is also its main requirement, is the following: *giving the user the possibility to express semantic assertions about a Web resource.*

Usability. One of the most important features that have supported the wide diffusion of current collaborative tagging systems is the handiness of the process of tagging; every user can immediately tag a Web resource using one or more keywords. We have considered the importance of this aspect trying *not to excessively increase the cognitive weight of the semantic tagging process.* It is obvious that we need a greater amount of information to be provided by a user in order to specify the intended meaning of a tag, but we have paid attention to organize and graphically arrange the interactions in order to make the semantic tagging process as fast as possible.

Motivation. Moreover we must consider *user's motivation to produce semantic assertions.* A critical aspect related to the diffusion of our system is the possibility of experimenting concrete advantages in information organization and accessibility, when adopting this new way of tagging. For this reason we have laid great emphasis also on the completeness and the availability of added value information organization and search features.

6.4.2 Main user interaction patterns

Starting from the system requirements just analyzed, we can define the structure of the principal interactions between our semantic tagging system and its typical users.

First of all a generic user can access our system from two fundamental different perspectives. He may use the system only *as a search engine performing a semantic assertion search* in order to find relevant Web resources references; this is a passive exploitation of the semantic information collected by the system, meaning

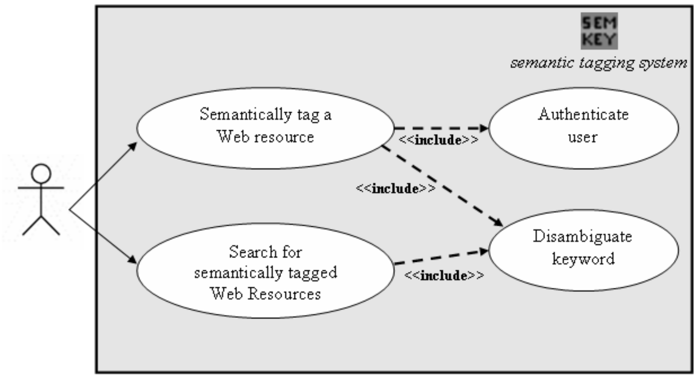


Figure 6.8: System UML use case diagram.

that the user accesses to the contents already present in the system without giving any contribution to its enrichment. In this way the user takes advantage of only a part of all the possibilities offered by our system, not exploiting one of its fundamental aspects: the social component. On the other end, the user, after having completed a registration phase, may *authenticate himself and access his personal area.* Thus he can exploit all the functionalities of SemKey. He can semantically tag Web resources of interest producing semantic assertions, manage his collection of resources and semantic assertions and organize them. Moreover, through his tagging activity, every user gives his contribution to the enrichment of the informative data collected by the system thus making searches possibly more effective. We can graphically schematize the described typical user-system interactions through the UML use-case diagram in Figure 6.8.

6.4.3 Global architecture of the system

Our system architecture is based on *three main modules*: two server-side resident components and a client side one. The main functionalities provided by each module are:

- **Semantic tagging manager** (client side): this module is intended to be strictly integrated in user browsers so as to allow a fast process of semantic tagging in order not to alter the usual Web browsing activity of a common user. In this way while using our semantic tagging system defining semantic assertions and sending them to the server, we aim not to introduce any change in the diffused browsing interaction patterns;

- **Sense disambiguation module** (server side): this module provides access to all information and services needed during the lexical form disambiguation process; it mainly supports the client in the choice of the intended concept described by a lexical form collecting the different meanings and thus allowing the definition of a semantic assertion;
- **Metadata store and access module** (server side): this is the principal module of our system. It mainly stores and provides Web access to all collected semantic tagging information. It is also responsible of the users' management.

In Figure 6.9 we represent SemKey high-level modules just described.

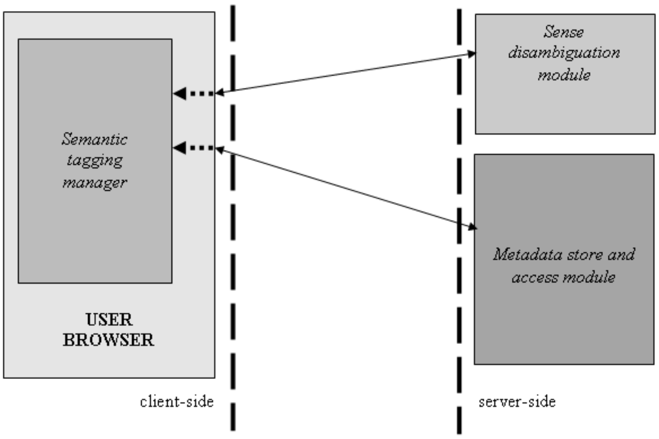


Figure 6.9: SemKey high-level modules.

When a user browses the Web and visits a resource of interest, he can decide to semantically tag it. In Figure 6.10 we show the sequence of interactions that characterize the production of a semantic assertion. The user activates the 'Semantic tagging manager' that retrieves the URL of the resource and allows the user to select a tag (or lexical form) (1). If the user isn't still logged in SemKey, logging credentials are requested in order to identify him; they are validated interacting with the 'Metadata store and access module' (2). After the authentication phase is successfully completed, the user will be driven in the choice of the intended meaning of the selected tag. Interacting with the tagging Web APIs of del.icio.us [del] and Yahoo My Web 2.0 [MyW] the 'Semantic tagging manager' retrieves and shows the user the most popular tags concerning the selected resource, in order to provide possible suggestions (3). Once the user has chosen a tag, it will be sent to the 'Sense disambiguation module' in order to receive a list of all possible concepts that can be referred using that tag (4). The user selects the intended meaning of his tag and the specific property of the Web resources to describe: thus he formulates a semantic assertion. It is sent to the 'Metadata store and access module' to be stored (5). Then the 'Semantic tagging manager' ends its execution and the user can continue his browsing (6).

6.4.4 Main organizational issues

In this section we analyze the basic organizational issues faced when structuring the modules which constitute our system. We discuss and motivate every adopted solution, but we also describe possible improvements and future scenarios.

The structure of Sense disambiguation module: WordNet and Wikipedia exploitation.

The 'Sense disambiguation module' represents the core of our system; it is devoted to support the client-side 'Semantic tagging manager' during the process of semantic tagging and it is responsible for

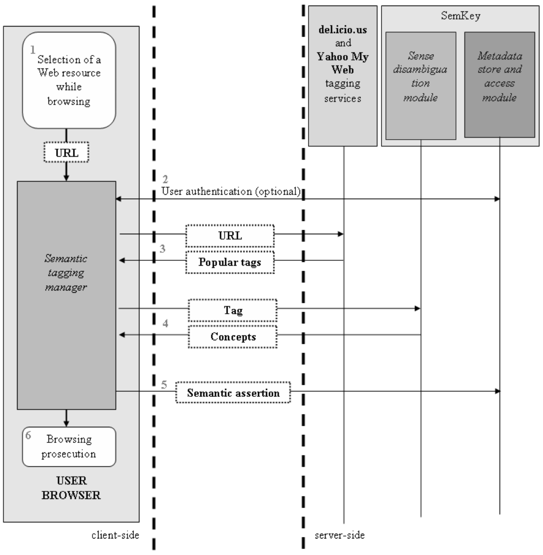

SemKey
1
Selection of a Web resource while browsing
URL
del.icio.us and Yahoo My Web tagging services
Sense disambigua tion module
Metadata store and access module
2
User authentication (optional)
Semantic tagging manager
URL
3
Popular tags
Tag
4
Concepts
5
Semantic assertion
6
Browsing prosecution
USER BROWSER
client-side
server-side

Figure 6.10: Modules interaction to produce a semantic assetion.

the collection of available meanings of user tags. As stated before, in this initial version of SemKey we have decided to explore the semantic content of WordNet [Wne], and Wikipedia [Wika]. During the disambiguation process, the 'Sense disambiguation module' accesses the Web interfaces of WordNet and Wikipedia to collect the meanings of the typed tag. In particular, given a tag we consider:

- **in WordNet**, all synsets which the tag belongs to;
- **in Wikipedia**, the description of the meaning of the tag or the different meanings associated to a polysemous tag through its disambiguation page.

The 'Sense disambiguation module' selects for every concept two information:

- an **URI** which identify the concept;
- a short textual **description** or gloss of the concept.

In what follows, we describe in more detail the general functioning of the sense disambiguation module.

Wikipedia and WordNet are both accessible through a Web interface; we can ask for the description of a concept identified by a string (tag). If the string associated to a particular concept is contained in these Web resources, we respectively get the following information:

- **Wikipedia** : an HTML Web page describing the concept or, if there is more than one concept associated to the tag string (polysemy), an HTML Web page of disambiguation with all available meanings of the tag, each one identified by a short description and the link to the URL of the Wikipedia page dedicated exclusively to it;

- **WordNet** : an HTML Web page containing the list of all available concepts (synsets) associated to the tag, each one described by a short gloss and referenced by an URL reference.

The 'Sense disambiguation module' is an aggregator of available WordNet and Wikipedia meanings of a tag. To support this task we have defined a particular elaboration sequence to be execute when we need to disambiguate a tag:

1. The tag disambiguation request is sent to the 'Sense disambiguation module', usually by the 'Semantic tagging manager' or by the user browser; the request includes the tag to disambiguate;

2. The 'Sense disambiguation module' retrieves from Wikipedia and WordNet, exploiting their HTML Web Interfaces, the Web pages associated to the tag (in Wikipedia, if the tag considered is a polysemous one, the module will receive a disambiguation page);

3. The available meanings of the tag and the URL associated to each one of them are extracted from Wikipedia and Wordnet responses, through appropriate XSLT transformations of

the XML normalized documents. The collected information is aggregated and a list of couples of reference URLs and respective short concept description is created;

4. The list of concepts is sent back to the 'Semantic tagging manager' or to the user browser.

This sequence of interactions is graphically represented in Figure 6.11.

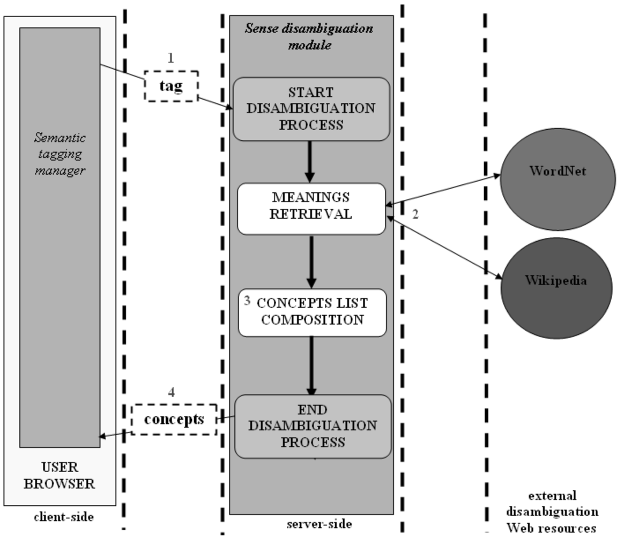

Semantic tagging manager
USER BROWSER
client-side
1
tag
4
concepts
Sense disambiguation module
START DISAMBIGUATION PROCESS
MEANINGS RETRIEVAL
2
3 CONCEPTS LIST COMPOSITION
END DISAMBIGUATION PROCESS
server-side
WordNet
Wikipedia
external disambiguation Web resources

Figure 6.11: Tag disambiguation module implementation and usual interactions.

Compared to WordNet, Wikipedia contents is often more difficult to manage to disambiguate a tag because of its relaxed organizational structure that doesn't provide many facilities to support this task.

6.4.5 Semantic assertion model

Another relevant issue is represented by the organization of the set of data stored during the semantic tagging of a Web resource.

In a generic collaborative tagging system as del.icio.us a user can associate a tag to a resource without specifying the relation type. Normally the tag represents the topic of the resource but this is not always true and this semantic information will be lost. A solution could be to force the user to explicate the kind of relation for each tag.

Starting from the analysis of the different kinds of tags managed by existing collaborative tagging systems, we have decided to manage only *three different relations*:

1. **hasAsTopic** : this relation will be used to describe the topic of the resource such as book, Web design, sport, politics, cars, animal, medicine, etc.;

2. **hasAsKind** : this relation will be used to characterize the kind of informative content of the resource such as blog, application, mashup, podcast, official Web site, streaming, video, e-commerce, Web API, etc.;

3. **myOpinionIs** : this relation concerns all subjective opinions such as cool, funny, interesting, boring, amazing, expensive, boring, etc..

The choice of the right relation to connect a concept to a particular resource is left to the user. In this way the model of a semantic assertion is a particular type of RDF triple (see Figure 6.6).

6.4.6 Semantic search patterns

When a user searches for relevant resources, he must specify the structure of one or more *generic semantic assertions*; they are semantic assertions defined without referring to a particular resource. Each of them specifies a concept that describes a particular characteristic (or property) of the resource to find. All the resources that are described by the set of generic semantic assertions specified by the user are considered to form search results.

For instance, the user could ask the system to find all 'blogs' (property: kind of resource) which deal with 'Web design' (property: topic of resource) and are reputed to be 'interesting' (property: personal opinion); 'blog', 'Web design' and 'interesting' are disambiguated lexical forms, referring to specific concepts. The search parameters just described are composed of three generic semantic assertions. The user could specify one or more semantic assertions.

6.4.7 Exploitation of WordNet and Wikipedia net of relations.

Those just described represent only the basic search capabilities and content structuring possibilities that our system offers. A possible relevant improvement could be obtained considering all the nets of relations that could connect the concepts used to support the disambiguation of lexical forms or could relate two or more different tag lexical forms. This further informative content is usually present in lexical resources.

In the first development phase of our system, we have decided to exploit the disambiguation information provided by the lexical resource *WordNet* [Wne]. In WordNet, the meaning and the lexical forms used to refer to a particular concept are connected by a set of 18 different kinds of relations. Some of these are very specific and have been introduced in order to exploit the semantic information available with an originally distinct purpose: text mining and information extraction. However, other relations could be exploited to further enrich search capabilities and structured exploration of contents. For example, *the hyponymy/hypernymy relations* that represent the hierarchical specialization / generalization of concepts may be used to suggest, during the disambiguation of lexical forms, all their hyponyms or hypernyms in order to better define the level of precision adopted by the user; in this way we can solve or at least reduce the basic level of precision problem, mentioned before. Moreover we can allow users to extend the coverage of their search including all the hyponym concepts of those related to particular chosen concept; in a similar way we can suggest users to choose one of the hyponyms of a disambiguated tag to better specify the search parameters. He can also substitute a disambiguated tag with one of its hypernyms in order to eventually increase search coverage.

For instance, if a user wants to find all the resources tagged with ’automobile’, after the choice of the intended meaning for this word, he could examine all the concepts that are hyponyms of this concept: ’jeep’, ’coupe’, ’station wagon’, etc. so as to extend the search coverage including all resources tagged with a least one of the hyponyms or to further refine his search replacing, for example, ’car’ with ’jeep’ and increasing the level of precision adopted. Part of the considered WordNet subsumption hierarchy of concepts is schematized in Figure 6.12.

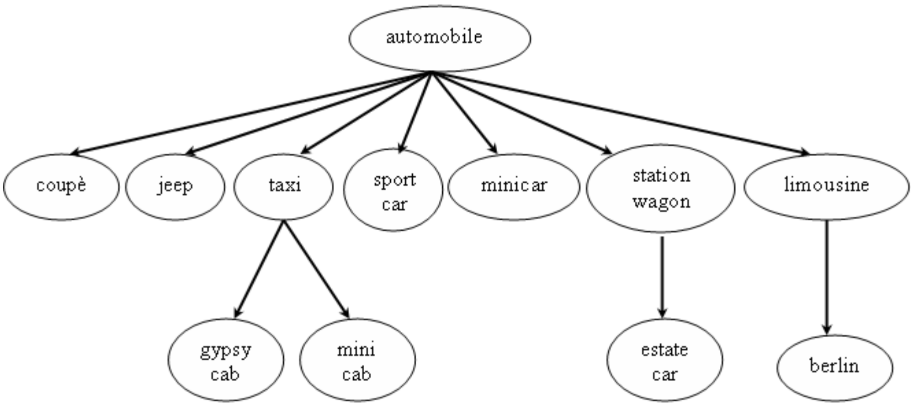


Figure 6.12: Part of WordNet hierarchy of concepts referred to the automobile world.

Another exploitable WordNet’s relation is *the meronym or ’part of’ relation*. It connects a concept with other concepts which constitute its parts. For example, the tag ’automobile’, used to refer to a four wheels vehicle, has the following parts or meronyms: ’accelerator’, ’air bag’, ’auto engine’, etc. It could be useful to show all meronyms of a concept in order to help users to better structure and organize their search tag set.

When we analyze *Wikipedia* [Wika] and its semantic concept references, we should consider that it is not a coherent lexical resource, but a collaboratively edited encyclopedia. Also Wikipedia provides a sort of content categorization system: *the Wikipedia categories.* They are collaboratively edited and managed and don't constitute a hierarchical structure; they form a direct graph. Every category could be included in one or more general ones, and sometimes there are also cyclic inclusions, even if editors are explicitly advertised to avoid such a situation. All those categories constitute a sort of specialization / generalization structure similar to that previously described speaking about WordNet, with more relaxed constraints. We can consequently exploit this added informative content in a way similar to that described considering WordNet, in order to improve search completeness. Besides the category structure, Wikipedia contains a highly dense net of simple interdocument references and every concept description or encyclopedia entry presents a collection of related Web resources which could be exploited to provide the user with useful links suggestions in order to deeply examine a concept.

6.5 Detailed system modules architecture

Considering the main high-level modules of our system, their interactions and the fundamental organizational issues faced when specifying their architectural structure, in Figure 6.13 we detail the internal organization of each of them.

The 'Semantic tagging manager', implemented as a browser extension, can directly interact with del.icio.us [del] and Yahoo My Web 2.0 [MyW] Web APIs to retrieve popular tags suggestions.

The 'Sense disambiguation module' can be accessed directly from the Web browser when the user must single out a specific concept considering a particular tag, in order to support SemKey search functionalities; this module can be also queried by the 'Semantic tagging manager' during the formulation of a semantic assertion in order to point out a particular concept.

The 'Metadata store and access module' can be accessed by the 'Semantic tagging manager' in order to save one or more semantic assertions, by a request to SemKey Web APIs or by the user browser in order to execute semantic searches or to manage the personal data of every user of our system.

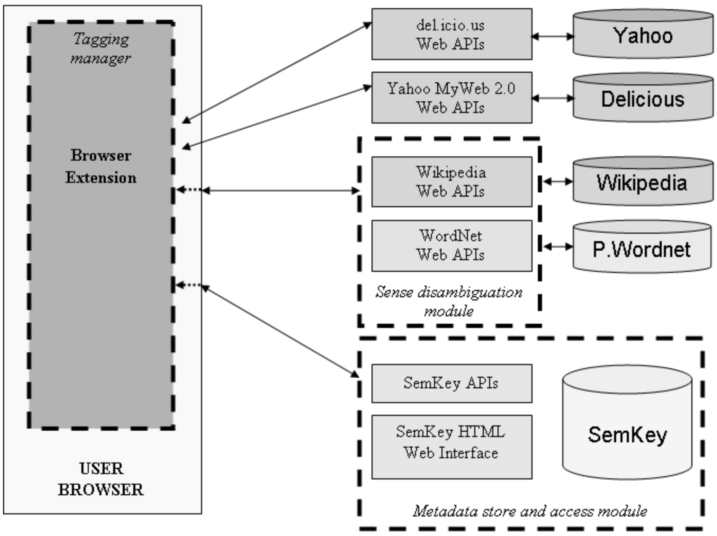


Figure 6.13: Detailed system modules' architecture.

6.5.1 Implementation and functioning

We describe some implementation details with some examples of SemKey in action, considering each one of its three main modules. The current version of the tool is available at http://www.semkey.org.

The Semantic Tagging Manager (STM)

STM is the client-side module of our system: it must support the user in the semantic tagging process (choice of the concept, starting from a lexical form, and the relation) and *maintain a high usability of our system.* We have implemented it as a Mozilla Firefox extension [Moz].

When the plug-in is installed, a multi-coloured button is added to the user interface of the browser. It is used to add one or more semantic assertion to the current resource (URL) displayed on the browser by activating a dialog window as shown in Figure 6.14.

If the user is not still logged in the system, he is requested to type his logging credentials (username and password) (2), interacting with the 'Metadata store and access module' so as to validate them. After the log-in phase is successfully completed, the user will be driven in the composition of the semantic assertion.

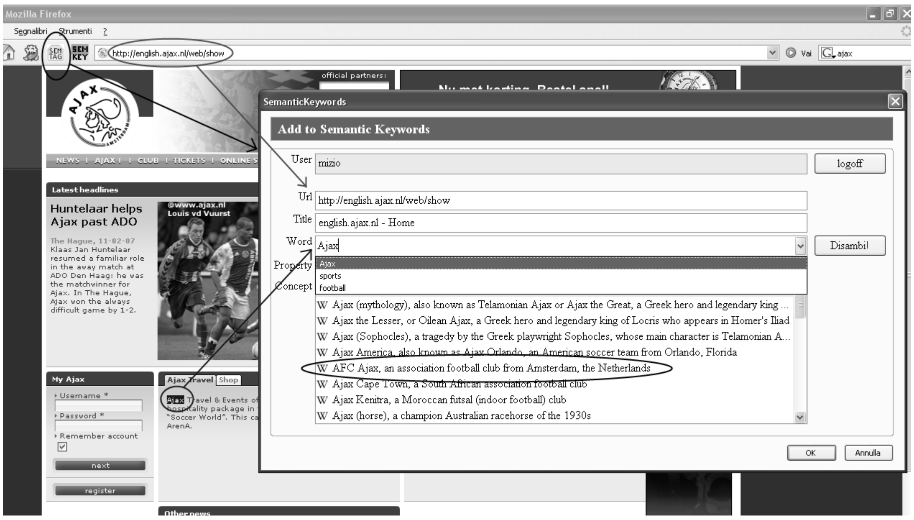
The STM proposes initially some tags corresponding to the most popular ones used by del.icio.us users to annotate the current resource. The user can select one of these tags or insert a new one and the relative relation (by default it is selected the 'hasAsTopic' relation).

Immediately the STM answers with a list of available meanings. Once selected the intended meaning of the considered tag, the semantic assertion is completed and STM will save it sending all data to the 'metadata store and access module'.

The Sense Disambiguation Module (SDM)

This module has to support the process of disambiguation of the tag chosen by the users. SDM gathers the different meaning of a particular lexical form by exploiting the available concepts of WordNet and Wikipedia. To carry out this goal, the SDM filters the web pages of these two lexical resources producing a list of the collected meaning associated to the lexical form. This list is serialized in order to compose a JSON array [JSO] with all collected couples of concept URLs and respective short concept descriptions; this array is sent back as the reply to the STM.

If Wikipedia and Wordnet provided some suitable Web APIs to access their content, we could simplify the SDM.



Add to Semantic Keywords

User: mizio [logoff]

Url: http://english.ajax.nl/web/show

Title: english.ajax.nl - Home

Word: Ajax [Disambi!]

Property

Concept

- Ajax
- sports
- football

- W Ajax (mythology), also known as Telamonian Ajax or Ajax the Great, a Greek hero and legendary king ...
- W Ajax the Lesser, or Oilean Ajax, a Greek hero and legendary king of Locris who appears in Homer's Iliad
- W Ajax (Sophocles), a tragedy by the Greek playwright Sophocles, whose main character is Telamonian A...
- W Ajax America, also known as Ajax Orlando, an American soccer team from Orlando, Florida
- W AFC Ajax, an association football club from Amsterdam, the Netherlands
- W Ajax Cape Town, a South African association football club
- W Ajax Kenitra, a Moroccan futsal (indoor football) club
- W Ajax (horse), a champion Australian racehorse of the 1930s

[OK] [Annulla]

Figure 6.14: Semantic tagging manager dialog window.

The Metadata Store and Access Module (MSAM)

This module provides storage functionalities to save and retrieve all semantic annotations. It is also responsible for the users management. All these features are available through a Web based HTML interface.

User Management. In our system each user must be registered in order to be identifiable; This allows us to manage his personal data and his tagging metadata and to support him with additional system functionalities.

Semantic Annotation. The main goal of our system is the collection of semantic assertions produced by the semantic tagging activity. Every semantic assertion is generated by a particular *user* in a precise *moment.* All these data are stored by the 'Metadata store and access module' as the outcome of semantic tagging activity.

User oriented views. When we speak about user oriented views, we mean all the available ways that a registered user, after his authentication, can exploit to interact with the system and visualize his personal profile data and his tagging metadata. Here is a list of the views implemented in our system:

- **Visualization of user semantic tagging metadata** :
    - *my Web resources* view : all the Web resources semantically tagged by the user ordered by date, with all the associated semantic assertions;
    - *my semantic assertions* view : all the semantic assertions made by the user ordered by referred concept and property (every semantic keyword is a link to the next view);
    - *my Web resources tagged with* view : all the Web resources semantically tagged with one particular concept (is a list of Web resource links ordered by date).
- **Deletion of a semantic assertion from a web resource** (*delete a tag* view).
- **Visualization / partial modification of user personal profile data** (*my profile* view);

Generic search-oriented view. This section includes all the available options that a generic user, authenticated in the system or not, can use to execute a semantic search among the collacted metadata:

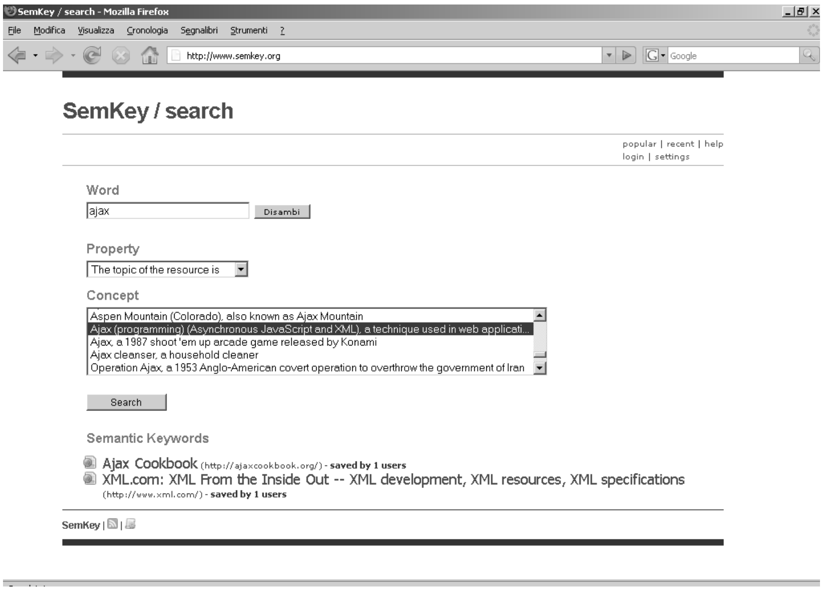
- *basic search* view : search all Web resources semantically tagged with one or more generic semantic assertion:
    - the user chooses one or more lexical forms;
    - every lexical form is disambiguated interacting with the 'sense disambiguation module' and retrieving its possible meanings; in this way the user can specify the intended concept, choosing between the multiple meanings presented;

- once a concept has been chosen, the user can select a particular property that links the Web resources he wants to find to the concept, thus defining a generic semantic assertion;
- SemKey, interacting with the 'Metadata store and access module', will retrieve all resources matching the set of generic semantic assertions previously defined.

Figure 6.15 shows an example of the *basic search* view system interface; the user has chosen the word 'ajax' and, among the list of concepts retrieved to allow its disambiguation, has selected the concept of 'AJAX (programming) (Asynchronous JavaScript and XML), a technique used in Web applications...'. Then selecting the property 'The topic of the resource is' has formed a generic semantic assertion, requesting to find all Web resources that speak about the Asynchronous JavaScript and XML Web programming technique. SemKey shows a list of all the semantically tagged resources that match the previously specified parameters.

6.6 Evaluation: semantic vs. syntactic tagging

The introduction of the possibility to formulate semantic assertions, disambiguating the meaning of tags can face or at least reduce the greatest part of the problems analyzed in section 6.2, related to current collaborative tagging systems. Our tagging system architecture just described is mainly intended for this purpose, as a first and improvable effort to produce added-value semantic tagging metadata. The main concept that underlies and supports our idea of semantic tagging and constitutes the basis of the architecture of our system is the following: *giving users the possibility to easily define semantic assertions, specifying the meaning of tags in a simple and usable, but also senseful way, trying to identify an efficient support to define shared semantics and to efficiently use this added informative value.* The main consequence of the introduction


SemKey / search - Mozilla Firefox
File Modifica Visualizza Cronologia Segnalibri Strumenti ?
http://www.semkey.org
Google

SemKey / search

popular | recent | help
login | settings

Word

ajax

Property

The topic of the resource is

Concept

Aspen Mountain (Colorado), also known as Ajax Mountain
Ajax (programming) (Asynchronous JavaScript and XML), a technique used in web applicati...
Ajax, a 1987 shoot 'em up arcade game released by Konami
Ajax cleanser, a household cleaner
Operation Ajax, a 1953 Anglo-American covert operation to overthrow the government of Iran

Semantic Keywords

- Ajax Cookbook (http://ajaxcookbook.org/) - **saved by 1 users**
- XML.com: XML From the Inside Out -- XML development, XML resources, XML specifications (http://www.xml.com/) - **saved by 1 users**

Figure 6.15: Example of semantic search.

of semantics is *a new organization of tagging metadata: they are no more a collection of strings, but a set of semantic assertions each of them referring to a specific concept.*

Consulting del.icio.us' help page, we can extract the following definition of a tag: 'Tags are one-word descriptors that you can assign to your bookmarks. They're a little bit like keywords but nonhierarchical'. As a result, currently the tag set is an unordered collection of freely chosen keywords, assigned to some resource (URLs in this case). Figure 6.16 represents the situation just described of an unstructured set of strings.

When we formulate semantic assertions we must consider concepts, specifying the meaning of tags: as a consequence the tag space will be differently organized to provide support to concept referencebility. In what follows, we refer to this new organization of the tag space as the **concept space**. There are two main entities that constitute a concept space: *concepts* and *lexical forms*. Concepts are abstract referable meanings, eventually described ref-

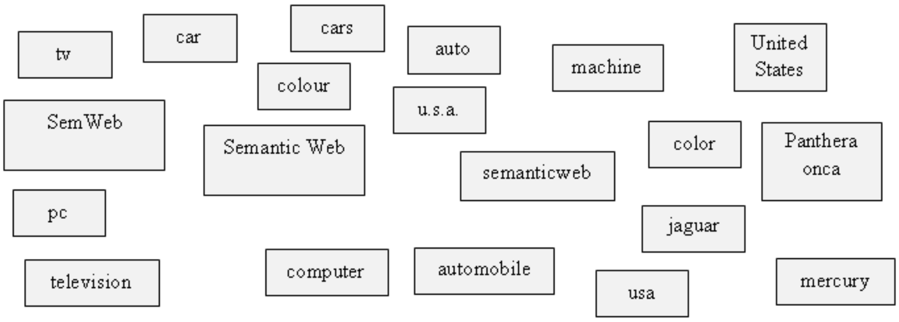


Figure 6.16: Unstructured tag space in current collaborative tagging system.

erencing a particular Web resource or through a short textual description; lexical forms are generic strings composed by one or more words. Every concept has one or more associated lexical forms; these strings constitute the string set through which a concept is usually referred to during the definition of a semantic assertion. All strings associated to a concept represent its synonyms, but also different lexical forms used to refer to the same meaning like common misspelling errors or alternate spelling. On the other end, every lexical form can be associated to more than one concept representing cases of polysemy. In Table 6.5 the main components of this new organization of the tag space are graphically represented.

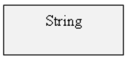
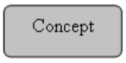
	
Lexical form	**Concept**

Table 6.5: Main components of the semantic tagging space.

Every lexical form could be associated with one or more concepts, constituting two kinds of possible connection schemata, shown in Figure 6.17 and 6.18.

When a single concept is represented by multiple associated lexical forms (see Figure 6.17), these ones may represent:

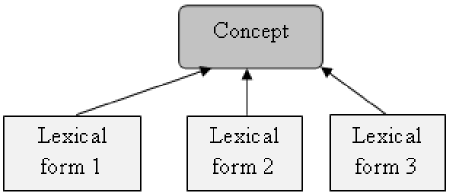


Figure 6.17: A single concept represented by multiple associated lexical forms.

- Synonyms;
- Misspelling errors;
- Alternate spelling;
- Acronym;
- Etc.

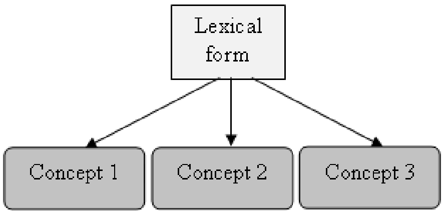


Figure 6.18: A single lexical forms which represents multiple meaning (concepts).

When a single lexical forms represents multiple meaning or concepts (see Figure 6.18), we are in presence of polysemy.

The considerable set of *problems caused by polysemy*, analyzed in section 6.2, could be solved considering semantics, thanks to the introduction of concepts and the associations between concepts and lexical forms that characterizes a concept space. During a semantic assertion based search, the user of the semantic tagging system can choose the right meaning intended for every tag. Using a small number of generic semantic assertions, each of them referred to a particular concept, we can increase the precision of our search results because we will collect only those resources related to a specified meaning, overcoming the ambiguities that arise from polysemy. Moreover the recall of the system is considerably improved because we collect all resources tagged with other kinds of lexical forms used to identify the same concept.

Also the *synonymy problem* is solved by the introduction of semantics; when we define a generic semantic assertion we refer to a concept, besides the related lexical form which represents only a mean to access to a specific conceptualization. In this way, we can overcame the current partitioning of the relevant search results subsequent to the possibility to access only to those ones tagged with the lexical form typed in the search tag set, previously analyzed.

Similarly all those problems related to every different lexical form which could be associated to a concept are solved or at least reduced: *misspelling errors*, *alternate spellings*, *relations between names*, *adjectives and verbs referring to the same meaning*, *notation differences*, *multi-word tags* and so on.

In what follows we describe possible solutions to some of the problems noticed in section 6.2, related to the examples previously provided. In particular we show an example of concept space organization (textually described and visually represented by a graph in Figure 6.19). The considered concept space contains the following concepts:

- **Concept 1** : the different acronyms and punctuation used to refer to the United States of America ('usa', 'u.s.a.', 'United Sates');

- **Concept 2** : the different words or abbreviations used when a user refers to a personal computer ('pc', 'computer');

- **Concept 3** : the different synonyms usually used when we speak about an automobile ('car', 'auto', 'automobile', 'machine'); 'cars' is a plural form (other lexical form);

- **Concept 4 and 5** : the lexical form 'mercury' presents multiple meaning (polisemy); it is related to the nearest planet to the Sun in the Solar System (concept 4) or to a chemical element (concept 5);

- **Concept 6, 7 and 8** : also the lexical form 'jaguar' has three different meaning represented by three conceptualizations; moreover the Concept 7 ('A large felid (animal) native to South and Central America') presents another multi-word lexical form, the scientific designation of the jaguar: 'Panthera onca';

- **Concept 9** : there are three possible multi-word or single word lexical forms used to refer to the concept of Semantic Web; the lexical forms 'semanticweb' or 'semweb' are possible abbreviations or different notations referring to the same concept;

- **Concept 10** : the different adopted spellings of the word 'color' ('color' and 'colour').

When the user wants to find all resources that describe Jaguar cars, he must only specify the intended meaning of the tag 'jaguar' so as to form a semantic assertion, without the need to disambiguate it using other tags and all the relate drawbacks. The system will collect all resources tagged by every other user selecting the same meaning.

Similarly, when the user wants to look for every resource speaking about cars, the Semantic Web or a personal computer, defining the generic semantic assertion he will select the concept referred and therefore he will retrieve all resources semantically tagged by assertions containing the same concept even if specified by synonyms ('auto' and 'machine' are synonyms of 'car'), other lexical forms ('cars' is another lexical form, the plural form that refers to the concept of automobile), abbreviations ('pc' is an abbreviation of the word computer) or other notations ('semWeb' is a compact notation and 'Semantic Web' a multi-word notation, both used to

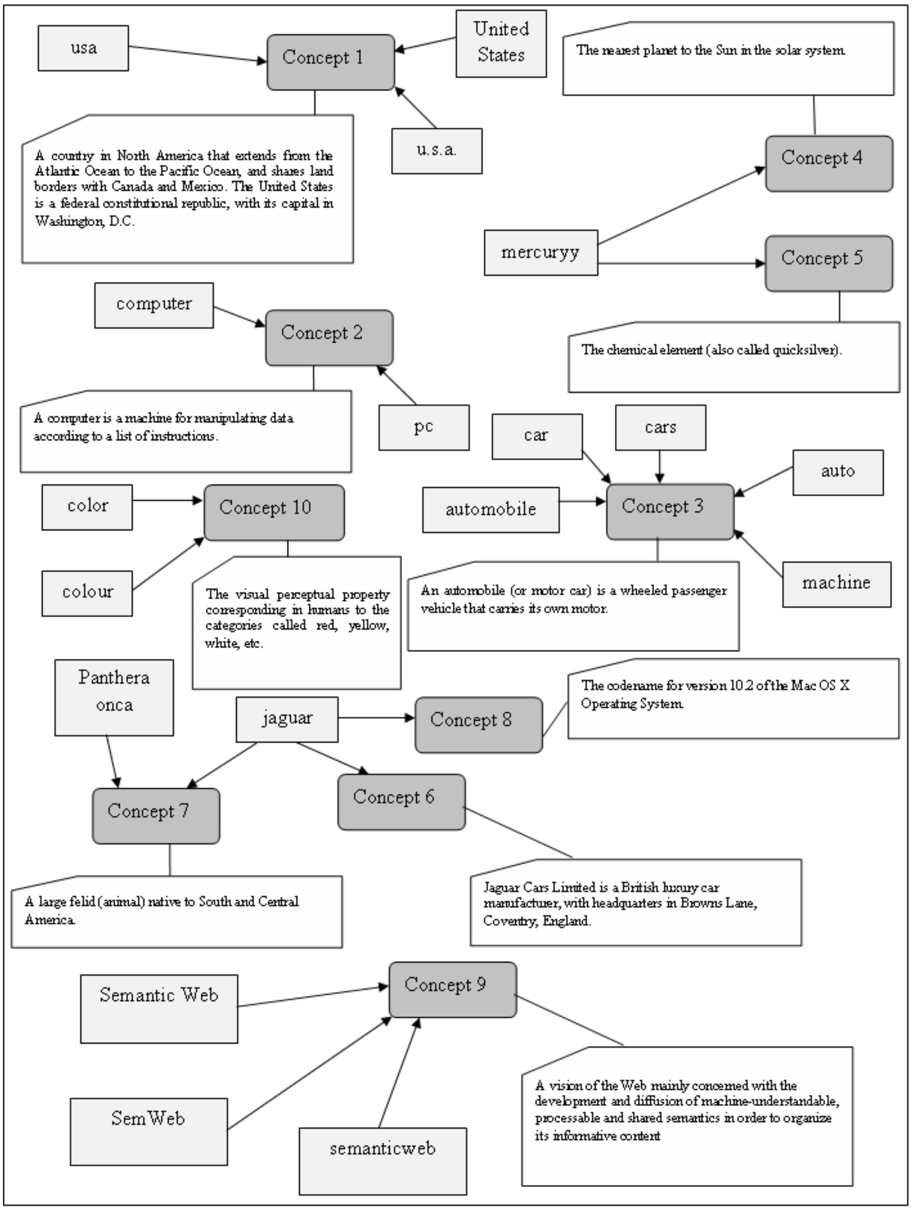

usa
Concept 1
United States
The nearest planet to the Sun in the solar system.
A country in North America that extends from the Atlantic Ocean to the Pacific Ocean, and shares land borders with Canada and Mexico. The United States is a federal constitutional republic, with its capital in Washington, D.C.
u.s.a.
Concept 4
mercuryy
Concept 5
computer
Concept 2
The chemical element (also called quicksilver).
A computer is a machine for manipulating data according to a list of instructions.
pc
car
cars
auto
color
Concept 10
automobile
Concept 3
colour
The visual perceptual property corresponding in humans to the categories called red, yellow, white, etc.
An automobile (or motor car) is a wheeled passenger vehicle that carries its own motor.
machine
Panthera onca
jaguar
Concept 8
The codename for version 10.2 of the Mac OS X Operating System.
Concept 7
Concept 6
A large felid (animal) native to South and Central America.
Jaguar Cars Limited is a British luxury car manufacturer, with headquarters in Browns Lane, Coventry, England.
Semantic Web
Concept 9
SemWeb
semanticweb
A vision of the Web mainly concerned with the development and diffusion of machine-understandable, processable and shared semantics in order to organize its informative content

Figure 6.19: Example of the structure of the concept space.

refer to the Semantic Web concept, maybe by different communities of users).

During the tagging activity, in case of misspelling or typing errors there are two possibilities of interaction: if the misspelled lexical form is already associated to the intended meaning, the user will simply select it to disambiguate his tag. If there are no meanings associated to the typed lexical form, the user will be driven by the system to reconsider the lexical form in order to eventually notice and correct the typing mistake (or to specify a new lexical form inside the system and associate it to a concept). This is an attempt to begin to solve also to the problem often referred to as metadata ecology [Ros01].

In conclusion, a further important aspect of our system is represented by *the possibility to extend the concept space using other kinds of connection or categorization rules in order to enrich its informative content.* For instance we can group the concepts in categories, or connect different concepts through particular useful relations like those typically used in lexical resources; we can simply define 'relatedTo' relations or more complex connections like specialization/generalization of concepts or part/whole between concepts. All those data could be added as an extension to theconcept space; their addition or their use to extend search possibilities should not be seen as a fundamental requirement, but should represent an optional chance to deeply exploit the semantic structure and usefulness of our tagging system and its semantic net of concepts.

An external Web entity could provide some sort of additional content or extension in order to better organize and connect concepts so as to increase search possibilities; for instance, a Web site could distribute its mapping between Wordnet synsets and our collection of concepts so as to exploit, where available, the Wordnet synsets' relations. On the other end, another Web site could provide its own organizational structure of some concept of interest in order to support its internal search possibilities or to define an alternative and generic way to explore and share its content. The interoperability of such a system could be greatly improved through the use of RDF as the standard representation language for every extension. Of course every system could be capable or not to support and exploit one specific extension to the concept space. For example, considering the addition of ’relatedTo’ and ’seeAlso’ relations, a search system may show the user also other suggested resources when performing a specific tag based search, increasing the completeness of his search experience. In the graph in Figure 6.20 we represent one example of an added useful search relations set. In the right sides is showed the concept space; in the left side is represented one possible external set of further concept to concept relations, for instance of ’relatedTo’ relations (the red lines). Through an external entity in this case we have added further informative content to our collection of concepts, stating that concept 1 is related to concepts 3 and 4, and concept 2 is related to concept 5. This information could be used to improve the user search possibilities and the informative content of the concept space.

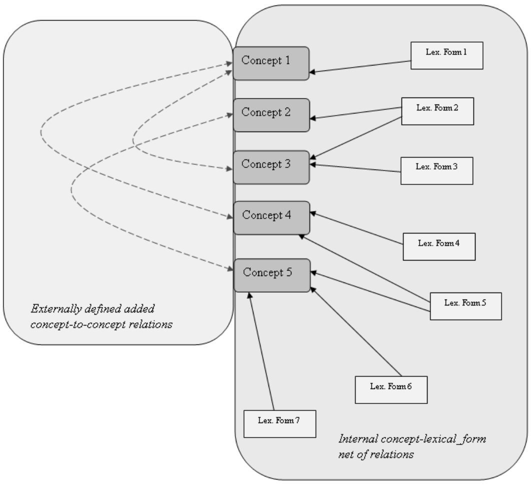


Figure 6.20: Example of a possible externally-defined extension of the concept space.

Since now we have analysed semantic tagging activity concerning a global domain. Recently, the advantages provided by tagging activity have been introduced also in enterprise networks; IBM has announced its version of an internal social bookmarking system: Doager [DMK05]. The exploitation of SemKey in specific knowledge domains represents another important potential field of application. Indeed, our semantic tagging system can be used for defined collections of concepts in order to describe a particular domain of interest. We think that future works could concern the possibility to exploit our semantic tagging tool as a corporate knowledge management and organizational support; it can support the organization and improvement of the accessibility to shared information like internal collections of documents or, in general, any huge amount of data which needs to be collaboratively organized. Many domain specific concepts collections are currently available: for instance MeSH [MeS], the National Library of Medicine's controlled vocabulary thesaurus is a terminological medical reference widely used and that could be adopted as a specific tagging reference. The analysis of this possibilities constitutes an interesting new semantic tagging application scenario.

Conclusions

In this thesis we have proposed a new approach for managing and exploiting semantic resources. The paradigm of distributed and interoperable semantic resources has largely been discussed and explored, and we have studied new methods and techniques for its practical realization. Overcoming the limits of singular resources required a change in the very basic assumptions on the design, creation, maintenance and distribution of knowledge resources and for this purpose interesting suggestions come from the emerging paradigm based on the notions of cooperation, collaboration and social knowledge determination. This paradigm subtends the practice of groups asynchronously producing works together through individual contributions in the so-called collaborative authoring.

On the basis of these researches we have developed a semantic resources manager prototype, called LexFlow[TMB+06], based on a distributed three-layer architecture (see Chapter 2). The higher layer is built on XFlow[MTM05], a framework for cooperative management of XML resources (see Chapter 3). This cooperative layer (see Chapter 4) is intended as an overall environment where all the modules implemented in the lower layers can be integrated in a comprehensive workflow of human and software agents. The middle layer hosts some applications that exploit the semantic shared repositories. One of these applications, the so-called MultiWordNet[STM+06] Service (MWS) allows to mutually enrich wordnets in a distributed environment (see Chapter 5). MWS can be proposed as a prototype of a web application that supports the Global WordNet Grid[gwn] initiative. The lower layer consists of a sort of grid of local services implemented as a virtual repository of XML databases residing at different locations and accessible through web services. Basic software services are also necessary, such as an UDDI server for the registration of the local wordnets and web services dedicated to the coherent management of the different versions of WordNet referred to by databases. In this work we have been concentrated on the description of the cooperative layer and the middle layer.

In order to demonstrate the possible use of these semantic resources we have also explored a new way to tag Web resources based on semantic concept and we have developed a semantic collaborative tagging system called SemKey. By analyzing the fundamental weak points of existing social tagging systems, we have deduced that most of them are referable to the absence of any semantic support in tagging activity. To solve, or at least simplify these problems we propose to substitute actual keywords or tags for a new kind of semantic-aware metadata: semantic assertions. They don't consist of simple strings related to a particular resource like existing tags; each semantic assertion describe a specific property of a resource. It associates a concept to a resource specifying the semantics of their relation. One or more different strings, called lexical forms, can be used to identify a particular concept; the set of strings related to a concept includes synonyms, different spelling or misspelling errors and all other possible lexical forms used to express a particular meaning. The activity of describing resources formulating semantic assertions is referred to as semantic tagging. We have implemented a semantic collaborative tagging system: SemKey. It allows experimenting the improved search efficiency and effectiveness and the new information access and organization patterns introduced thanks to semantic tagging activity.

The basis of our idea of semantic tagging is the availability and completeness of a global collection of concepts and lexical forms in order to specify and univocally reference the concepts of semantic assertions; both WordNet and Wikipedia have been used in order to test their possible support to this tasks. We have explored their main organizational features: WordNet contains a rich set of parts of speech and a strongly structured net of relations between them, but it lacks many data useful to support proper names disambiguation and it is not collaboratively edited; Wikipedia is an encyclopedia so its contents are composed mainly by a very rich set of names along with their extended descriptions. Wikipedia has strong proper names coverage; it is also continuously updated, but lacks a structured set of relations between the concepts described, even if its documents are interconnected by a huge number of links: at present, only the system of Wikipedia categories is available as an attempt to provide some sort of relaxed structure to its informative content. Besides Wikipedia and WordNet we must mention an early project OmegaWiki [Ome]; it is attempting to build a free socially-edited multilingual thesaurus; it organizes concepts and terms adopting a structure that seems capable of supporting the disambiguation and concept referenceability needed by semantic tagging. In parallel with the growing of OmegaWiki informative content, future works should be oriented to better explore its possibilities of cooperation with semantic tagging systems.

Summarizing, we have suggested a new semantics-improved tagging pattern and we have developed SemKey, a semantic tagging system, in order to combine semantic technologies with the collaborative tagging paradigm in a way that can be highly beneficial to both areas.

Index

Bibliography

[AAF$^+$03] Roventini A., Alonge A., Bertagna F., Calzolari N., Cancila J., Girardi C., Magnini B., Marinelli R., Speranza M., and Zampolli A. Italwordnet: building a large semantic database for the automatic treatment of italian. *Linguistica Computazionale*, 18-19:745–791, 2003.

[ACLG02] Lerina Aversano, Gerardo Canfora, Andrea De Lucia, and Pierpaolo Gallucci. Integrating document and workflow management tools using xml and web technologies: A case study. In *CSMR '02: Proceedings of the 6th European Conference on Software Maintenance and Reengineering*, page 24, Washington, DC, USA, 2002. IEEE Computer Society.

[AMM01] P.Lazzareschi A. Marchetti, S. Minutoli and M.Martinelli. A system for managing documents in a step by step process. In *XML World Euro Edition*, 2001.

[AT00] Vineet Kakani Shremattie Jaman Anand Tripathi, Tanvir Ahmed. Implementing distributed workflow systems from xml specifications. Technical report, Department of Computer Science, University of Minnesota, 2000.

[BC02] Richard V. Benjamins and Jesus Contreras. Six challenges for the semantic web. 2002.

[BCC$^+$99] L. Baresi, F. Casati, S. Castano, M. G. Fugini, I. Mirbel, and B. Pernici. Wide workflow development methodology. In *WACC '99: Proceedings of the international joint conference on Work activities coordination and collaboration*, pages 19–28, New York, NY, USA, 1999. ACM Press.

[Bec06] Dave Beckett. Semantics through the tag. Conference presentation XTech 2006: Building Web 2.0, Amsterdam, The Netherlands, May 2006. Yahoo! Inc.

[Bid03] Matt Biddulph. A semantic web shoebox - annotating photos with rss and rdf. Technical report, 2003.

[BLTO01] Harrdelr J. Barners-Lee T. and Lassila O. The semantic web. *The Scientific American*, May:34–43, 2001.

[bpw] W3c semantic web best practices and deployment working group - web site. http://www.w3.org/2001/sw/BestPractices/.

[CM06] Danah Boyd Marc Davis Cameron Marlow, Mor Naaman. Position paper, tagging, taxonomy, flickr, article, toread. Technical report, Yahoo! Research Berkeley 1950 University Avenue, Suite 200 Berkeley, CA 94704-1024 - UC Berkeley School of Information 102 South Hall Berkeley, CA 94720-4600, 2006.

[CN05] Soria C. Calzolari N. A new paradigm for an open distributed language resource infrastructure: the case of computational lexicons. In *Knowledge Collection from Volunteer Contributors. Papers from the 2005 AAAI Spring Symposium*, 2005.

[CRHC05] Cui-Xia Weng Hsiang-Ping Lee Yong-Xiang Chen Chu-Ren Huang, Chun-Ling Chen and Keh-Jiann Chen. The sinica sense management system: Design and implementation. *Computational Linguistics and Chinese Language Processing*, 10-4:417–430, 2005.

[CRHL04] Ru-Yng Chang Chu-Ren Huang and Shiang-Bin Lee. Sinica bow (bilingual ontological wordnet): Integration of bilingual wordnet and sumo. In *LREC2004*, 2004.

[Cro95] W. Bruce Croft. What do people want from information retrieval? D-Lib Magazine, 1995.

[CTZ02] Paolo Ciancarini, Robert Tolksdorf, and Franco Zambonelli. Coordination middleware for xml-centric applications. In *SAC '02: Proceedings of the 2002 ACM symposium on Applied computing*, pages 336–343, New York, NY, USA, 2002. ACM Press.

[DB06] Canyang Kevin Liu David Booth. Web services description language (wsdl) version 2.0. http://www.w3.org/TR/wsdl20-primer/, 2006.

[del] del.icio.us tagging system web site. http://del.icio.us/.

[DMK05] Jonathan Feinberg David Millen and Bernard Kerr. Social bookmarking in the enterprise - ibm. *ACM Queue*, vol. 3, no. 9, 2005.

[DMW06] Olena Medelyan David Milne and Ian H. Witten. Mining domain-specific thesauri from wikipedia: A case study. Technical report, Department of Computer Science, University of Waikato, 2006.

[DPR] J. Daudé, L. Padró, and G. Rigau. A complete wn1.5 to wn1.6 mapping.

[Dub] Dublin core metadata initiative web site. http://dublincore.org/.

[ewn] Euro wordnet project - official web site. http://www.illc.uva.nl/EuroWordNet/.

[Fra05] Enrico Franconi. Introduction to semantic web ontology languages - formalising ontologies. Keynotes @ Semantic Wb Application Conference 2005, December 2005. Faculty of Computer Science, Free University of Bozen-Bolzano, Italy.

[Fut06] Joe Futrelle. Harvesting rdf triples. Technical report, Natioanl Center for Supercomputing Applications 1205 W. Clark St., Urbana IL 61801, US, 2006.

[GAMM93] Christiane Fellbaum-Derek Gross George A. Miller, Richard Beckwith and Katherine Miller. Introduction to wordnet: An on-line lexical database. Technical report, 1993.

[gde] Google docs & spreadsheets - web site. http://docs.google.com/.

[GH02] G. Gulrajani and D. Harrison. *SHAWEL: Sharable and interactive Web-Lexicons.* European Language Resources Association, 2002.

[GH05] Scott A. Golder and Bernardo A. Huberman. The structure of collaborative tagging systems. Technical report, Information Dynamics Lab, HP Labs, 2005.

[GHS95] Dimitrios Georgakopoulos, Mark F. Hornick, and Amit P. Sheth. An overview of workflow management: From process modeling to workflow automation infrastructure. *Distributed and Parallel Databases*, 3(2):119–153, 1995.

[Gru93] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowl. Acquis.*, 5(2):199–220, 1993.

[GT98] Dimitrios Georgakopoulos and Aphrodite Tsalgatidou. Technology and tools for comprehensive business process lifecycle management. In Asuman Dogac, Leonid Kalinichenko, Tamer Ozsu, and Amit Sheth, editors, *Proceedings of the NATO Advanced Study Institute on Workflow Management Systems*, NATO ASI Series F, pages 324–363, 1998.

[GT06] Marieke Guy and Emma Tonkin. Folksonomies: Tidying up tags? *D-Lib Magazine*, Volume 12 Number 1:6, January 2006.

[Gua98] N. Guarino. Some ontological principles for designing upper level lexical resources, 1998.

[gwn] The global wordnet association - official web site. http://www.globalwordnet.org/.

[HST] Murray Maloney-Noah Mendelsohn Henry S. Thompson, David Beech. Xml schema part 1: Structures second edition. http://www.w3.org/TR/xmlschema-1/.

[ILC03] Nancy Ide, Alessandro Lenci, and Nicoletta Calzolari. Rdf instantiation of isle/mile lexical entries. In *Proceedings of the ACL 2003 workshop on Linguistic annotation*, pages 30–37, Morristown, NJ, USA, 2003. Association for Computational Linguistics.

[JC] Murata Makoto James Clark. Relax ng specification. http://www.oasis-open.org/committees/relax-ng/spec-20011203.html.

[JC99] Steve DeRose James Clark. Xml path language (xpath) version 1.0. http://www.w3.org/TR/xpath, 1999.

[JDW06] Rudi Studer John Davies and Paul Warren. *Semantic Web Technologies.* John Wiley & Sons, Ltd, 2006.

[JMB06] Roland Merrick-T. V. Raman-Micah Dubinko Leigh L. Klotz John M. Boyer, David Landwehr. Xforms 1.0

(second edition). http://www.w3.org/TR/xforms11/, 2006.

[JMM04] Juan Llorens Jorge Morato, Miguel ngel Marzal and Jose Moreiro. Wordnet applications. Technical report, Dept. Computer Science, Universidad Carlos III, Madrid, Spain - Dept. Library Science, Universidad Carlos III, Madrid, Spain, 2004.

[JSO] Json (javascript object notation) - web site. http://json.org/.

[Kay07] Michael Kay. Xsl transformations (xslt) version 2.0. http://www.w3.org/TR/2007/REC-xslt20-20070123/, 2007.

[KMK02] Rupa Krishnan, Lalitha Munaga, and Kamalakar Karlapalem. Xdoc-wfms: A framework for document centric workflow management system. In *Revised Papers from the HUMACS, DASWIS, ECOMO, and DAMA on ER 2001 Workshops*, pages 348–362, London, UK, 2002. Springer-Verlag.

[Kro] Ellyssa Kroski. The hive mind: Folksonomies and user-based tagging. Blogsite.

[KRSRR97] G. Kappel, S. Rausch-Schott, S. Reich, and W. Retschitzegger. Hypermedia document andworkflow management based on active object-oriented databases. In *HICSS '97: Proceedings of the 30th Hawaii International Conference on System Sciences*, page 377, Washington, DC, USA, 1997. IEEE Computer Society.

[LA03] Pirrelli V. Lenci A., Montemagni S. Chunk-it. an italian shallow parser for robust syntactic annotation. *Linguistica Computazionale*, 16-17:353–386, 2003.

[LA05] R. Lambiotte and M. Ausloos. Collaborative tagging as a tripartite network. Technical report, SUPRATECS, Universit de Lige,B5 Sart-Tilman, B-4000 Li'ege, Belgium, 2005.

[Lit97] Kenneth C. Litkowski. Computational lexicons and dictionaries. Technical report, CL Research, Damascus, Maryland, USA, 1997.

[LSG05] Ajay Mallya Luke Simon, Ajay Bansal and Thomas D. Hite Gopal. A universal service-semantics description language. Technical report, Gupta, Department of Computer Science, University of Texas at Dallas, 2005.

[Mat] Wolfram mathworld - the web's most extensive mathematics resource. http://mathworld.wolfram.com/Inclusion-ExclusionPrinciple.html.

[Mat04] Adam Mathes. Folksonomies - cooperative classification and communication through shared metadata. Technical report, Computer Mediated Communication - Graduate School of Library and Information Science - University of Illinois Urbana - Champaign, 2004.

[MeS] Medical subject headings - official web site. http://www.nlm.nih.gov/mesh/meshhome.html.

[MH06] Katharina Siorpaes Martin Hepp, Daniel Bachlechner. Harvesting wiki consensus - using wikipedia entries as ontology elements. Technical report, Digital Enterprise Research Institute (DERI), University of Innsbruck - Florida Gulf Coast University, Fort Myers, FL, USA, 2006.

[Mit03] Nilo Mitra. Soap version 1.2 part 0: Primer. http://www.w3.org/TR/soap12-part0/, 2003.

[MK06] Max Volkel Markus Krotzsch, Denny Vrandecic. Wikipedia and the semantic web. Technical report, Institute AIFB, Unviersity of Karlshrue, Germany, 2006.

[MKSW06] Mark-Jan Nederhof Marc Kemps-Snijders and Peter Wittenburg. Lexus, a web-based tool for manipulating lexical resources. In *LREC2006*, 2006.

[MM06a] George Macgregor and Emma McCulloch. Collaborative tagging as a knowledge organisation and resource discovery tool. Technical report, Centre for Digital Library Research, Department of Computer & Information Sciences, University of Strathclyde, 2006.

[MM06b] Choukri K. Friedrich J. Maltese G.-Mammini M. Odijk J. Ulivieri M. Monachini M., Calzolari N. Unified lexicon and unified morphosyntactic specifications for written and spoken italian. In *LREC 2006: 5th International Conference on Language Resources and Evaluation.*, 2006.

[Moz] Mozilla developer center - plugin. http://developer.mozilla.org/en/docs/Plugins.

[MPV02] Sabine Schulte im Walde Massimo Poesio, Tomonori Ishikawa and Renata Vieira. Acquiring lexical knowledge for anaphora resolution. In *LREC, Las Palmas*, 2002.

[MR03] Bindi R. Goggi S. Monachini M.-Orsolini P. Picchi E. Rossi S. Calzolari N. Zampolli A. Marinelli R., Biagini L. The italian parole corpus: an overview. *Linguistica Computazionale*, 16-17:401–421, 2003.

[MTM05] Andrea Marchetti, Maurzio Tesconi, and Salvatore Minutoli. Xflow: An xml-based document-centric workflow. *Lecture Notes in Computer Science*, 3806, 2005.

[MyW] Yahoo my web 2.0 apis reference. http://developer.yahoo.com/search/myweb/.

[N.06] Calzolari N. Technical and strategic issues on language resources for a research in-frastructure. In *International Symposium on Large-scale Knowledge Resources (LKR2006*, 2006.

[NC03] Alessandro Lenci Monica Monachini Nicoletta Calzolari, Francesca Bertagna. Standards and best-practice for multilingual computational lexicons. mile (multilingual lexical entry). Technical report, Istituto di Linguistica Computazionale - CNR, 2003.

[OGGM] A. Oltramari, A. Gangemi, N. Guarino, and C. Masolo. Restructuring wordnet's top-level: The ontoclean approach.

[Ome] Omegawiki web site. http://www.omegawiki.org/Main_Page.

[owl] Owl web ontology language reference - w3c recommendation. http://www.w3.org/TR/owl-ref/.

[Pla] Platypus wiki - the semantic wiki wiki web. http://platypuswiki.sourceforge.net/.

[PVDOA98] Wim Peters, Piek Vossen, Pedro D&#237;ez-Orzas, and Geert Adriaens. Cross-linguistic alignment of wordnets with an inter-lingual-index. pages 149–179, 1998.

[Rdf] W3c resource description framework web site. http://www.w3.org/RDF/.

[RFMSA06] Laurent Romary, Gil Francopoulo, Monica Monachini, and Susanne Salmon-Alt. Lexical markup framework (lmf): working to reach a consensual iso standard on lexicons. In *LREC2006*, 2006.

[RN03] Gola E. Calzolari N. Del Fiorentino M.C. Ulivieri M. Rossi S. Zamorani N. Ruimy N., Monachini M. A computational semantic lexicon of italian: Simple. *Linguistica Computazionale*, 18-19:821–864, 2003.

[Ros01] Lou Rosenfeld. Folksonomies? how about metadata ecologies? Blog article, January 2001.

[Sch] Schematron, iso/iec fdis 19757-3. http://www.schematron.com/iso/dsdl-3-fdis.pdf.

[She05] Rashmi Shena. A cognitive analysis of tagging (or how the lower cognitive cost of tagging makes it popular). Blogsite, September 2005.

[SJHB96] Hans Schuster, Stefan Jablonski, Petra Heinl, and Christoph Bussler. A general framework for the execution of heterogenous programs in workflow management systems. In *COOPIS '96: Proceedings of the First IFCIS International Conference on Cooperative Information Systems*, page 104, Washington, DC, USA, 1996. IEEE Computer Society.

[Smi04] Gene Smith. Folksonomy: social classification. Blog article, August 2004.

[SS02] Korth Silberschatz and Sudarshan. *Database system concepts*, chapter 22, pages 852–853. Mc Graw Hill, 2002.

[ssw] Social software (form english wikipedia). http://en.wikipedia.org/wiki/Social_software.

[STM$^+$06] Claudia Soria, Maurizio Tesconi, Andrea Marchetti, Francesca Bertagna, Monica Monachini, Chu-Ren Huang, and Nicoletta Calzolari. Towards agent-based cross-lingual interoperability of distributed lexical resources. In *Proceedings of the Workshop on Multilingual Language Resources and Interoperability*, pages

17–24, Sydney, Australia, July 2006. Association for Computational Linguistics.

[SWH06] Nigel Shadbolt and Massachusetts Institute of Technology Wendy Hall, University of Southampton Tim Berners-Lee. The semantic web revisited. *IEEE INTELLIGENT SYSTEMS*, MAY/JUNE:69–101, 2006.

[THS05] Ben Lund Tony Hammond, Timo Hannay and Joanna Scott. Social bookmarking tools (i) - a general review. *D-Lib Magazine*, Volume 11 Number 4, 2005.

[TMB$^+$06] Maurizio Tesconi, Andrea Marchetti, Francesca Bertagna, Monica Monachini, Claudia Soria, and Nicoletta Calzolari. Lexflow: a system for cross-fertilization of computational lexicons. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 9–12, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

[Tol02] Robert Tolksdorf. Workspaces: A web-based workflow management system. *IEEE Internet Computing*, 6(5):18–26, 2002.

[TS01] Robert Tolksdorf and Marc Stauch. Using xsl to coordinate workflows. In *Kommunikation in Verteilten Systemen*, pages 127–138, 2001.

[Vos04] Piek Vossen. Eurowordnet: a multilingual database of autonomous and language-specific wordnets connected via an inter-lingualindex. *Int J Lexicography*, 17(2):161–173, 2004.

[Vos06] Jacob Voss. Collaborative thesaurus tagging the wikipedia way. Technical report, Wikimedia Detushland e.V., 2006.

[w3c] World wide web consortium official web site. http://www.w3.org/.

[Wei05] David Weinberger. Tagging and why it matters. Technical report, Harvard Berkman Center for the Internet and Society, 2005.

[Wika] English wikipedia web site. http://en.wikipedia.org/wiki/.

[Wikb] Wikipedia statistics english. http://stats.wikimedia.org/EN/TablesWikipediaEN.htm.

[Wne] Princeton wordnet web site. http://wordnet.princeton.edu/.

[wnw] W3c rdf/owl representation of wordnet - editor's draf. http://www.w3.org/2001/sw/BestPractices/WNET/wn-conversion.html.

[Wro06] Luke Wroblewski. The web now: Social. Technical report, Yahoo! Inc., 2006.

[wsp] Sparql query language for rdf - w3c working draft. http://www.w3.org/TR/rdf-sparql-query/.

[you] Youtube - web site. http://www.youtube.com/.

[You06] Edward Yourdon. Web 2.0 mind-map. Technical report, 2006.

[ZXS06] Jianchang Mao Zhichen Xu, Yun Fu and Difu Su. Towards the semantic web: Collaborative tag suggestions. Technical report, Yahoo! Inc 2821 Mission College Blvd., Santa Clara, CA 95054, 2006.