Knowledge Management in a Wiki Platform via Microformats

Sergiu Dumitriu and **Marta Gîrdea** and **Sabin Buraga**
Faculty of Computer Science, "Alexandru Ioan Cuza" University
General Berthelot 16, Iaşi, Romania
{sergiu.dumitriu,marta,busaco}@info.uaic.ro

Abstract

The paper presents a conceptual solution and an implementation for acquiring, modeling, publishing, retrieving, reusing and maintaining knowledge within XWiki, an open source collaborative system. We propose a new microformat (*hLocation*) to denote georeferences (locations). The provided case-studies illustrate the facets of Semantic Web (metadata, microformats, ontologies) participating to the platform's enhancements.

Introduction

When advancing towards Semantic Web (Berners-Lee, Hendler, & Lassila May 2001; Daconta, Smith, & Obrst 2003; Shadbolt, Hall, & Berners-Lee 2006), the main obstacle is the effort of organizing the knowledge (content, metadata, ontological constructs) made by the content providers. In the current systems, the users must work with certain vocabularies, tagging entities and relations. The purpose of these processes is to make the data comprehensible not only for humans, but also for computers.

A possible solution is the use of a flexible collaborative platform which helps to transparently organize (meta)data for machine-comprehensibility.

Recently, the predominant type of collaborative Web-based systems is the wiki (Leuf & Cunningham 2001), which offers premises for creating the knowledge base on a certain domain by using Web technologies. A traditional wiki system provides a Web interface for (collaborative) content editing. Main facilities are the simplified syntax, the rollback mechanism, the (possibly) unrestricted access, several search functions, the support for uploading content. Wiki platforms are currently used for different purposes, such as encyclopedias, software development, project management, personal knowledge management, content management, collaborative writing, and many others (Schaffert, Gruber, & Westenthaler 2005).

To achieve significant knowledge acquisition, a modern wiki must support user collaborative tools and, more important, must allow attaching metadata to the concepts and relations established between the involved concepts.

Knowledge Management

In this section, we tackle the problem of knowledge flow in collaborative environments. We refer to the six aspects identified in the Advanced Knowledge Technologies (AKT) objectives (Shadbolt & O'Hara 2004), and we describe a model which makes use of semantic web principles and integrates them into collaborative environments, without requiring, from the contributing authors of the information managed within the platform, any particular effort to introduce metainformation or any knowledge of semantic Web languages. The paper emphasizes the types of metadata we consider relevant in such a collaborative environment, proposes a way of expressing such information, and gives an example of automatic reasoning based on this structured information (metadata and microformats).

The Problem

Currently, most online content is created only with the visual result in mind, disregarding any semantic structure of the (X)HTML source. Table-based layout, images replacing text, extensive use of presentational tags like `<font>` and `<center>`, make it even harder for Web crawlers to retrieve information, in the same way a human agent perceives it. Automatic extraction of the Web documents' semantics is a very difficult task. This is why current search engines are still based on keywords and can not answer simple questions[1].

Web sites should aid this knowledge harvesting by structuring information in a semantic manner, in order to make knowledge available both for humans and computers. Because it is very unlikely for human agents to semantically organize information and to attach metadata, the underlaying Web platform should transparently offer knowledge management.

AKT identifies six challenges that concern the engineering and management of knowledge: acquiring, modelling, reusing, retrieving, publishing and maintaining knowledge (Shadbolt & O'Hara 2004).

[1] Several search engines provide answers to a few standard questions, such as "Population of Japan" or "Pope John Paul II's birthday".

Our Approach

In our vision, a non-intrusive knowledge management platform should have the following approach:

Acquiring The system should acquire knowledge, and not formatted text. We noticed that it can be difficult for people without education in the field to get used to concepts such as XML. Therefore, asking all users to markup the content they enter in a Website using RDF and OWL would fail. The fundamental concept of semantic Web is the triple (*entity* has *property* with *value*). The common perception is that the *entity* (an information fragment), although it belongs to a structured *class*, is a region of an on-line document, identified by an URI. Storing the data only as formatted text causes the loss of the semantic relations between information fragments. For example, in a "Call for Papers" document it is difficult to markup the relations between the event and its location or the involved persons (organizers, invited lecturers, regular participants).

A better solution would be to store the concepts only as structured information, and display their properties/relations in the document as needed. Acquiring such structured information can be performed using (X)HTML forms. Each form field identifies the entity and the property, and the user must assign the proper value.

Another way of information gathering involves obtaining selected (semi)structured data from external sources (RSS/Atom feeds, Web services, CGI scripts, RDF stores, databases).

Classes can also be defined on the basis of triple model. Each class property is defined starting from standard types (i.e., XML Schema datatypes: string, number, boolean etc., and reference), and form fields are used to adapt it by defining the property name and restricting the range.

Modelling Although each piece of information should be semantically modelled, this approach seems almost impossible in practice[2]. Our point of view is to rigorously express certain important information following the object-oriented paradigm.

To properly model the knowledge, some steps must be performed (Daconta, Smith, & Obrst 2003):

1. the *classes* needed to be used are identified;
2. the *properties* are defined;
3. the information regarding the *individuals* (class instances) is filled in via collaborative mechanisms by the involved users.

In contrast with closed-world approach (where all classes and properties are a-priori defined), XWiki permits adding and modifying classes and the respective properties at any moment.

[2]A solution could involved the disambiguation of the content via human language technologies.

Reusing Each entity is defined only once and attached to the mostly related document. The information can be globally accessed (e.g., the information about a user can be displayed in any page), according to certain permissions (for example, the users can edit a certain document fragment only if they belong to a given group). In order to encourage users to reuse the existing information, the system should provide an easy to use and understand syntax.

Retrieving Due to rigorous manner of structuring content, it should be easy to retrieve pieces of information by performing different queries. Supplementary, the system will be able to automatically aggregate, retrieve, and extract information on the basis of metadata and inter-object relations. For example, ad-hoc groups of users (including the relations of collaboration between them) can be obtained from certain metadata regarding their interests.

Publishing With the help of well-known metadata standards (such as DCMI – Dublin Core Metadata Initiative) and microformats (Khare & Çelik 2006) – e.g., hCard, hResume, XFN, and our proposal hLocation (see below), information can be properly organized for both human and computer access and can be rendered according to the user needs and preferences. For example, the news regarding the activities of a research group can be distributed and published in many forms by using hCalendar and RSS/Atom feeds. For a better semantic (re)use, the stored knowledge can be exported to RDF/OWL documents (all internal classes have corresponding OWL classes and the individuals can be exported as RDF triples).

Maintaining Because our approach is a wiki-like one, the knowledge can be easily maintained in a collaborative manner. The users can annotate, review, re-edit, and delete fragments of information as needed.

Conceptual Model for a Collaborative Wiki-like System

Entities and Relations

As it is not feasible to model any information, we must identify and define the most important concepts regarding a collaborative platform. General concepts for all wiki systems are: users (individuals and groups), documents (composed by content and metadata), events, and places. For publishing each of them, we make use of the existing metadata languages or we investigate the possibility of defining new ones.

Users This class denotes the concept of *user* and should describe the following attributes: name, address(es), affiliation, interests, location, age, contact information. For specific-purpose collaborative systems, this class can be refined in order to better reflect the user concept (e.g., an e-learning wiki can use `Student` class derived from the general user class to capture important knowledge about a student). We can adopt several well-known semantic

Web-based models such as FOAF (Friend Of A Friend), vCard (Dawson & Howes 1998), and – of course – RDF. At the presentation level, we use certain microformats like hCard, XFN (Extensible Friends Network), hResume, and hReview – details in (Microformats 2007).

Documents This class models the structure and the content of a given document available in different formats, managed by several users. Beside the document content, the platform should store metadata regarding the author(s), language(s), creation and modification dates, subject, keyword list or social tags, access rights, versions, etc. The proper model is the DCMI standard which can be embedded into the XHTML view of the document or in an external RDF file.

Events This class denotes an event concerning the community (for example, a group meeting, a conference, or a document release). Each event is mainly described by a title, a location, different temporal entries (i.e. start date, end date), relations to other events, etc. Event data can be automatically exported into machine-understandable formats like iCalendar and RSS/Atom. At the presentation level, hCalendar and hReview can be used.

Places The purpose of this class is to express knowledge about a given place via georeferencing. According to (Winter & Tomko 2006), the broad sense of *georeferencing* is associating information (such as documents, datasets, maps, biographical information, artifacts, images, specimens, directions etc.) to certain geographic locations through place names (i.e., toponyms), place descriptions (e.g., "five star hotel"), place relations (for example, "the hotel near conference location"), place codes (e.g., postal codes), or through geographical coordinates (geocode). These kinds of information are commonly employed by a collaborative platform. For this, a microformat can be used to denote all this information (see the following section).

All these entities are strongly inter-connected: documents relate to users, there are relations between people, places relate to events and vice-versa, users are interested by events/places, and so on.

A Microformat for Georeferencing

Context The Web documents can store different kinds of georeferences, usually included into the content, expressed by some natural language constructs, and addressed to the human reader. Natural language constructs require a proper understanding of some semantic and world knowledge to be shared between the content provider and the reader (for example, a postal address like "Berthelot Street, 16" can not easily processed by another application which expect "Berthelot:16" format for this address; even humans can have difficulties in interpreting such information).

A more interesting approach, in our context, is to embed georeferences in regular Web pages, with the help of microformats. Because microformats imply a certain structure, it can be easy to use them to aid computers to extract desired information in a given context and to aggregate content from many sources.

***hLocation* Microformat** *hLocation* is a simple, open, distributed format for representing georeferences, for example: museums, libraries, pubs, offices, banks, islands, home addresses, etc. that can appear in any Web content.

Following any other microformats design, the *hLocation* schema consists of the following:

- `hlocation` (root element) – denotes a georeference;
- `name` – name of the location (e.g., "Atlan City");
- `description` – an optional text with valid XHTML markup;
- `url` – an URI which provides more details about the given location;
- `email` – an email address attached to a georeference;
- `adr` – is an address, as specified by vCard (*post-office-box*, *street-address*, *extended-address*, *locality*, *region*, *postal-code*, *country-name*, *delivery-type*), but extended with some finer-grained fields: *street*, *number*, *block*, *floor*, *room*, and *sector*;
- `geo` – geographic latitude and longitude information: *latitude* and *longitude*;
- `tz` – time zone(s) associated with the location: *name*, *utc-offset* or *area*;
- relations with other resources via the `rel` attribute of a hyperlink: *part-of*, *contains*, *near*, *in-front-of*, *behind*, *above*;
- `rel-tag` – social tagging markups, for example: city, pub, museum, workplace, home, etc.;
- `image` – an image associated to the georeference; a further refinement could specify the point of view regarding the snapshot (e.g., the hotel as seen from above) or a given map (for example, a map provided by an external Web service).

Our approach provides a more detailed description of several attributes regarding locations (such as addressing, related locations in fuzzy terms, images) which can not easily be expressed by other microformats. Additionally, we use the most useful constructs provided by the already established microformats.

Example We use the following scenario: J.S. Bach, I. Calvino, and S. Dali are three artists who want to attend a conference in Atlan City, a location in Atlantis. The *hLocation* constructs embedded into the collaborative platform used by them are expressed by:

```
<div class="vevent hlocation">
<h2 class="adr">CFP:
  <span class="summary">Conference on Artistic
     Developments</span> at <a class="url name"
     href="http://www.places.org/Atlantis/Hall"
     >Hall of Mirrors</span>, in
  <span class="city">Atlan City</span>,
  <span class="country-name">Atlantis</span>,
  <abbr class="dtstart" title="20070511">
```

```
  May 11, 2007</abbr></h2>
<img src="hall.png" class="image"
     alt="The Hall of Mirrors" />
<h3>The conference will take place near
  <a href="http://www.places.org/Atlantis/MuseumArts"
     rel="near">Museum of Arts</a>.</h3>
</div>
```

Our Implementation

XWiki Overview

We based our implementation on the XWiki platform (XWiki 2006), a powerful wiki engine written using open source Java technologies. XWiki is a professional wiki, having many features needed for enterprise usage. It is also an application wiki, allowing development of applications with structured data and scripting right from the wiki interface. It can be adapted into almost any kind of web tool, from a simple presentation site, a personal wiki or blog, to a complex structured content repository or advanced collaborative application.

The power of XWiki lies in the multi-layered architecture, separating the platform into distinct, loosely coupled modules, based on the MVC (Model-View-Controller) design pattern.

Data Organization in XWiki

The information in XWiki is organized into documents, classes, objects, and attachments.

The *Document* is the entity that specifies the displayed content. It can contain raw text, wiki or (X)HTML markup, and interpreted scripts. Each document is described by properties (metadata), such as the document creator, content language or modification date – see above.

A document can be used to host a *Class* by attaching class properties, thus defining the class whose name is the name of the document. Classes model the structure of some target information stored in the wiki.

Objects are instances of classes, attached to the document they were created in. Certain document objects define structured metadata related to the document (for example, comments and page access rights), while other types of objects can be used to define the actual document content as a structured information unit, such as the entities expressing concepts such as users, groups, preferences or blog entries.

Attachments are files uploaded by users and whose content is stored in the database. The platform defines metadata for each attachment, such as upload time, file size or creator.

User Interface and Presentation

Editing information units in XWiki is aided by the simple wiki syntax, a friendly WYSIWYG editor, and various other AJAX-based tools. Structured documents can be built using a class definition, a template from which the document contents and attached objects are copied, and a sheet which transforms the document editing into a form completion task (in-place editing). This defines the Class-Template-Sheet pattern of structured data.

Rendering in XWiki is controlled by a number of Velocity template files. The Velocity code can use the XWiki API to manipulate the documents and their attached metadata (properties, objects, attachments), and can use the API that each plug-in defines. The content of a document is formatted by such templates for Web rendering, portlet integration, PDF or RSS/Atom feed generation, etc.

From Information Wiki to Knowledge Wiki

Context

XWiki handles internally a large range of well-structured information and metadata whose purpose regards mostly the user's benefit. However, this rich information is not fully valued. It is displayed as raw data, along with all the other contents of the documents, and cannot be automatically identified and used by a machine, despite the well-defined structure.

When advancing towards Semantic Web standards (W3C 2007) compliance, such information should be presented in machine-comprehensible vocabularies. This section describes the metamorphosis of the user friendly information wiki into a user and machine friendly knowledge wiki – more details in (Dumitriu, Gîrdea, & Buraga 2007).

Support for Metadata

DCMI (Dublin Core Metadata Initiative) For each document, the creator and the creation and modification date are usually displayed using plain HTML markup in a dedicated area of the page. To make this information semantically accessible, this information, along with other document properties, can be displayed using the DCMI vocabulary [dcmi] using meta-tags in the document header. The same information can be displayed using an external RDF document describing the document.

Microformats Some of the predefined classes of XWiki, namely the ones describing users and events, present well structured information units, for which microformats can be attached. Since microformats require only separating the different parts of the description and specifying the class attribute, adapting XWiki to display hCard and hCalendar requires minor modifications to the sheets that render user pages and event objects, with no impact on the visual layout of the information.

In the same manner, other microformats can also be used, like *adr*, *hResume*, or our proposed *hLocation*. Moreover, each presentation sheet can be organized in a semantic way, thus defining a custom microformat. For example, within an enterprise collaborative environment we can use a microformat concerning products and services (e.g., a not yet defined *hProduct*) and an entertainment multimedia platform can employ the use of a microformat about aural content (e.g., a not yet defined *hMusic*).

Using FOAF The user profile is displayed using plain markup, making it difficult for an agent to detect the fact that a page actually describes a person. For such information, FOAF elements can be expressed via RDF or RDFa constructs. With the help of different query techniques (i.e. SPARQL) and social tags, we can generate FOAF documents expressing specific groups of interest.

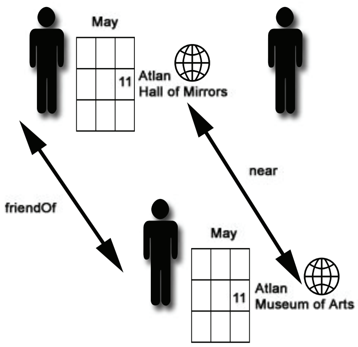

Figure 1: Different relations between two entities

Semantic Links Most semantic wiki systems provide support for link descriptions. For this purpose, we extended the Radeox (Radeox 2006) filter, which transforms the wiki syntax for links into (X)HTML anchors, to accept two more parameters, rendered as the content of the `title` and `rel` attributes, respectively. Such feature obviously allows defining links "with a meaning" (as an opposite to plain linking, which is almost exclusively used in hypertext). An advanced Web crawler can gather information about the linked resources (for example, in a page about a "book" resource, one can detect which of the links point to book chapters, which points to the author, the location of the publisher, etc.).

Regarding the georeferences denoted by *hLocation* microformat, we consider the relations established between georeferences are expressed by `rel` and `rel-tag` attributes (see the given example).

Folksonomy Support We defined a custom class to represent tags, which bind keywords to certain entities included into a XWiki document. Each user can attach tags to resources (for example, events, locations, people). This allows retrieving of similar pages, browsing information according to the tags, and obtaining better results for keyword-based search – in a manner adopted by social Web tools (e.g., del.icio.us, Flickr.com, etc.) – see (O'Reilly 2005) for details.

Case-Studies

Person Location Tracking via *hLocation* Using the previous scenario, all of three artists have expressed in their personal calendars the intention to visit the location (Atlan City, a city of Atlantis) on May 11, 2007. Two of them (S. Dali and I. Calvino) attend the Conference on Artistic Developments, while J.S. Bach is going to visit the museum near the conference location. Fact: S. Dali is a friend of J.S. Bach.

By using microformats and metadata embedded into the Web documents of the XWiki platform used by all three, an intelligent tool can conclude that S. Dali may meet J.S. Bach in Atlan City, on the given date.

Formalizing, the following rule can be made: if P_1 isFriendOf P_2 and $P_1(t_1)$ hasLocation L_1 and $P_2(t_2)$ hasLocation L_2 and t_1 equals t_2 and (L_1 near L_2 or L_1 equals L_2) then P_1 mayMeet P_2.

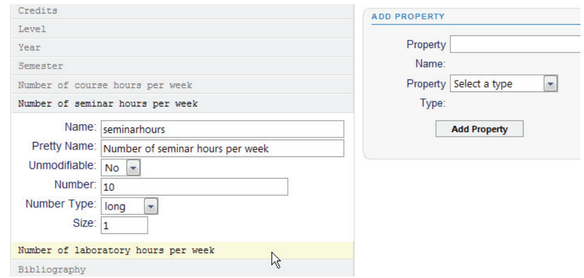

Figure 2: XWiki Web-based Class Editor

On the basis of metadata (FOAF) and microformats (hCard, hLocation, hCalendar), the system can extract needed data, can detect the above relations and can contact S. Dali and J.S. Bach in order to announce the other's presence in Atlan City and schedule a meeting via their iCalendar documents. Additionally, GoogleMaps Web services can be used to automatically enrich the content presentation with supplementary information about the location.

This solution is platform- and device-independent and uses only open standards and initiatives.

Editing and Managing Ontological Content This case study concerns the Web site (infoiasi.ro 2006) of our department. This Web site is built on the XWiki platform. The entities that participate to the academic life in a computer science department are persons (professors, students, and auxiliary personnel), specializations (undergraduates, graduates, and postgraduates of each age group), course curricula, and other types of resources (e.g., geographic locations, events). This concepts form a portal ontology, similar with AKT ontology (Shadbolt & O'Hara 2004), but more adequate for our purposes.

Each wiki entity is modeled as a class (for example, *Professor* derived from *User*, *Department*, *Generation*, *Course*, *OptionalCoursePackage*, etc.). Every class instance has different associated metadata exported as RDF statements. Several properties establish relations between different kind of objects (e.g., a *Professor* `teaches` a *Course* and a *Course* `isTaught` to a *Generation* of students). This approach helps us to offer a proper navigational support within the wiki and minimizes redundant data.

Supplementary, at the user level, a list of quick and related links, is automatically generated according to the document relations.

This internal class-oriented structure can be easily exported into RDF and/or OWL formats.

Related Work and Initiatives

Actual collaborative systems target large user communities and are mainly based on folksonomies and microformats. For example, flickr.com uses *hCard* in user profiles, Blogger.com and WordPress use *rel-tag*, and upcoming.org uses *hCalendar*. Concerning georeferences, the most relevant platform is Plazes.com (in beta stage).

Another direction of development is focused on small and specialized communities which use rigurous ontological models, such as MindSwap, BioPax, AKT, or pOWL.

Our intention is to unify both directions, emerging to a wiki-like collaborative platform which uses social Web technologies and trends (e.g., microformats, social tagging, REST, AJAX, etc.).

From the implementation point of view, several initiatives of wiki platforms which give support for expressing knowledge via current Semantic Web technologies are already available. Most of them are simply ontological editors in the wiki style, which can not be used by people without a prior knowledge on (Semantic) Web area. Semantic Web technologies are relatively difficult and, of course, can not be suitable for regular people to use them within a certain wiki as a collaborative platform or in the context of semantic desktop.

Most important initiatives are (Semantic Wiki 2006):

- *Rhizome* offers support for capturing and representing informal, human-authored content and is based on the actual Semantic Web languages, but requires the knowledge of XML and related technologies. Rhizome Wiki uses custom text formats to make writing semantic content easier, which can be managed by a RDF data access engine. Unfortunately, the rendered content does not contain embedded rich metadata or microformats and final users do not fully benefit of RDF-like stored knowledge.
- *Semantic MediaWiki* is a prototype built on PHP-based MediaWiki platform using an external RDF repository. A possible use of such a system is to facilitate knowledge acquisition;
- *IkeWiki* supports different levels of formalization (from informal texts to formal ontologies expressed in OWL) and a rich user interaction. IkeWiki uses the well-known Jena semantic platform, and presents to users links to (semantically) related pages.
- *OntoWiki* is a tool that integrates RDF triples into wiki textual content and simplifies the presentation and acquisition of instance data from and for end users in a generic way, in order to ease the ontologic content creation in a collaborative manner.

We noticed the majority of available semantic wiki systems can be used by specialists only and can not address the needs of regular communities to exploit the power of semantic Web technologies in a pragmatic real-world way.

Conclusions and Further Work

The paper presented a possible platform-independent Web-based solution for knowledge management in a wiki manner. Our proposed implementation uses the XWiki platform and gives support for knowledge representation and manipulation, aligned to current standards of the Semantic Web (RDF, FOAF, microformats, OWL).

Our system can be a viable solution regarding complex operations for a certain community of interest (such as an academic organization or an enterprise). For this purpose, we followed an engineering approach for resource management. We used several microformats to rigorously capture, represent and manipulate the knowledge available within a wiki, in a pragmatic manner, even at the desktop level (e.g., using certain tools able to manage microformats and to invoke Web services). Also, we modelled georeferences by using a new – more refined – microformat, called *hLocation*.

A further direction of research is the design of new microformats to capture knowledge about other types of resources. Another interest is the study of integrating search tools and automatic reasoning services within XWiki in order to enhance the user interaction.

References

Berners-Lee, T.; Hendler, J.; and Lassila, O. May 2001. Web Semantic. *Scientific American*.

Daconta, M.; Smith, K.; and Obrst, L. 2003. *The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management*. John Wiley & Sons.

Dawson, F., and Howes, T. 1998. vCard MIME Directory Profile. IETF RFC 2426: http://www.ietf.org/rfc/rfc2426.txt.

Dumitriu, S.; Gîrdea, M.; and Buraga, S. 2007. From the Information Wiki to Knowledge Wiki via Semantic Web technologies. In *Proceedings of CISSE 2007*. Springer-Verlag, Berlin. to appear.

infoiasi.ro. 2006. Website of The Faculty of Computer Science, "A.I. Cuza" University, Iaşi, Romania. Website: http://www.infoiasi.ro/.

Khare, R., and Çelik, T. 2006. Microformats: a Pragmatic Path to the Semantic Web. *Proceedings of the 15th International Conference on World Wide Web*.

Leuf, B., and Cunningham, W. 2001. *The Wiki way: Quick Collaboration on the Web*. Addison-Wesley.

Microformats. 2007. Microformats. Website: http://microformats.org/.

O'Reilly, T. 2005. What is Web 2.0 – Design Patterns and Business Models for the next Generation of Software. Webpage: http://www.oreillynet.com/lpt/a/6228.

Radeox. 2006. The Radeox Wiki Rendering Engine. Website: http://radeox.org/.

Schaffert, S.; Gruber, A.; and Westenthaler, R. 2005. A Semantic Wiki for Collaborative Knowledge Formation. In *Proceedings of Semantics 2005*.

Semantic Wiki. 2006. Semantic Wiki State of the Art. Webpage: http://wiki.ontoworld.org/index.php/Semantic_Wiki_State_Of_The_Art.

Shadbolt, N., and O'Hara, K., eds. 2004. *Advanced Knowledge Technologies – Selected Papers 2004*. Webpage: http://www.aktors.org/publications/selected-papers/.

Shadbolt, N.; Hall, W.; and Berners-Lee, T. 2006. The Semantic Web Revisited. *IEEE Intelligent Systems* 21(3).

W3C. 2007. The World Wide Web Consortium. Website: http://www.w3.org/.

Winter, S., and Tomko, M. 2006. Translating the Web Semantics of Georeferences.

XWiki. 2006. XWiki. Website: http://www.xwiki.org/.