Platypus Wiki: a Semantic Wiki Wiki Web

Stefano Emilio Campanini, Paolo Castagna, Roberto Tazzoli
{campanini, castagna, tazzoli}@users.sourceforge.net

Abstract

This article describes PlatypusWiki, an enhanced Wiki Wiki Web using technologies from the Semantic Web. Platypus Wiki offers a simple user interface to create Wiki pages including metadata according to W3C standards. It uses RDF, RDF Schema and OWL to manage the metadata and create ontologies. We present the essential features of what we have called a Semantic Wiki Wiki Web, showing how the existing Wiki Wiki Web can be improved and how we have implemented these features in Platypus Wiki. Platypus Wiki is a rapid and useful Personal Knowledge Management system, as well as a valuable tool to manage Communities of Practice.

1. Introduction

In this article, we present Platypus Wiki, a prototype of a Semantic Wiki Wiki Web, an enhanced Wiki Wiki Web using the RDF model and OWL vocabulary to represent metadata and relations between Wiki pages.

A Wiki Wiki Web, or simply Wiki for short, is a web site where users can contribute by adding content on any page. When a page is created other users can edit the content in a collaborative manner. Platypus Wiki extends the Wiki Wiki Web idea by adding metadata to Wiki pages and using RDF properties to represent the `meaning' of links and to provide the navigational layout.

Many web sites organize information using taxonomies with a hierarchical tree, while the RDF model represents structures as a directed labeled graph. Each node of this graph is an RDF resource, while an arc between nodes is an RDF property. Moving between nodes in a graph is different from moving in a hierarchical tree, as people can use different paths to reach the same node.

Platypus Wiki is an ongoing project enabling the collaborative editing of vocabularies and ontologies according to RDF Schema and OWL recommendations.

The first section of the paper discusses the Wiki Wiki Web. The key features are described, in particular the reason why Wikis are useful as Knowledge Management or Personal Knowledge Management systems, and how they can be used to manage a community. Various issues regarding Wikis are underlined, and possible solutions are proposed.

Next, Communities of Practice are introduced in terms of their importance in sharing information and creating new knowledge.

In the third section, the idea of a Semantic Wiki Wiki Web is described in detail. Platypus Wiki is a prototype implementation of this idea. The operation and features of the system are illustrated and the improvements with respect to standard Wiki systems highlighted.

In the conclusion, we outline our plans for possible future developments and new ideas.

2. Wiki Wiki Web

There is no definition of the Wiki Wiki Web that is shared by everyone. It is a discussion medium, a repository of ideas and a tool for collaboration. In a Wiki Wiki Web, anyone can add or edit a page using a simple syntax to write content. This allows anyone to participate and permits the site to evolve in a collaborative way.

The idea of the Wiki Wiki Web [1] and the first running implementation comes from Ward Cunningham who published the Portland Pattern Repository's Wiki in May 1995 [2]. “Wiki wiki” is a Hawaiian term for *quick* and is pronounced *wee-kee*. The name underlines the simplicity of a publishing system that is easy to learn and quick to use.

In his Wiki, Cunningham describes the earliest design principles of a Wiki Wiki Web. A Wiki should be open in the sense that if readers find a Wiki page incomplete or poorly organized they can change and complete it, even if they are not the original authors. The process is incremental, and pages can cite other pages even before they have been completed. In this way, a page author can express an interest in talking about a given subject.

The structure of the site is also open and evolves freely in the same way as the content. While standard web sites organize their content with a predefined

hierarchical structure, Wikis use an organic approach with an evolving graph structure. Wiki pages are nodes in the graph, while users can establish links between these nodes. A simple syntax has been proposed to allow anyone to publish, while hiding the HTML markup. This syntax creates links automatically. The name of a page is defined by the convention of writing two or more concatenated words with initial capital letters. When a page is sent from a server to clients, these particular words, called *WikiNames* (or *CamelWords*), are automatically replaced by links to that page. If the page does not exist, the system offers the possibility to create it at this time. The system thereby invites users to contribute to the Wiki by adding content, while at the same time organizing relations between new and existing pages. Content within the site can be viewed and reviewed by any user. Such a site in which anyone can edit any page could lead to a state of disorder, but in fact the opposite is true. After eight years of activity, Portland Pattern Repository is full of valuable content in thousands of pages and brings together a large number of readers and writers.

After the original Wiki implemented in Perl, many clones were released in different programming languages, as well as many public Wikis. One of the most valuable projects based on a Wiki engine is *Wikipedia* [3], a free-content, collaboratively developed multilingual encyclopedia. The project started in January 2001 first in English and then in many other languages. As of March 2004, the encyclopedia contained over 230,000 articles in English and over 300,000 articles in other languages.

Other examples of Wikis are *Apache Wiki*[1], *Java.net Wiki*[2], *Javapedia* and *ESW Wiki*[3]. All these initiatives started during 2003 and are used to manage Communities of Practice.

3. Communities of Practice

A virtual community is "a group whose members are connected by means of information technologies", while Communities of Practice are "groups of people who share a concern, a set of problems, or a passion about a topic, and who deepen their knowledge and expertise in this area by interacting on an ongoing basis" [4, 5].

The fundamental elements of a Community of Practice are: a domain of knowledge, a community of people and shared practices. The domain and community depend on the interests of the people involved. From a point of view of a Semantic Wiki Wiki Web, shared practices are developed online by editing Wiki pages. Use of a web-based, collaborative content management system allows people to share ideas and agree on essential concepts. With technologies borrowed from Semantic Web research, the concept can be represented in a way that can be understood by machines as well as people. In a Community of Practice, knowledge is developed within the community itself, and the practices represent the vocabulary and common agreements among a group people who try to explain their latent knowledge.

Communities of Practice are functional to a group of experts interested in a specific argument. The motivation of the people is essential, as is the support of the technology. Wikis can be seen as a supporting technology for a Community of Practice: they enable users to discuss around a page representing a concept, they adapt to situations in which knowledge changes quickly, and they do not impose any extra overhead on those who want to contribute.

In recent years, the rapid increase in information has brought the need to improve knowledge management tools. At the same time, the enormous wealth of information has increased the amount knowledge available to anyone and the subsequent complexity in managing this information. Furthermore, the half-life of knowledge is becoming shorter, so it is difficult for an individual to keep up-to-date on a given topic without the help of a community.

The community helps users to feel part of a greater project, and this is an important motivational factor which should not be ignored. Collaboration and knowledge are keys to success in business, organization and self-management.

A Wiki can be seen as a network of inter-related thoughts [6]. Wikis are "think tanks" that allow people to free their brains, to move around and to manage more information more quickly than usual.

4. Semantic Wiki Wiki Web

A Semantic Wiki Wiki Web is a Wiki improved with Semantic Web [7] technologies. In this section, we present the essential features of a Semantic Wiki Wiki Web and how it can be implemented. The implementation is written in Java, and we have called our prototype Platypus Wiki because the platypus is a strange animal which challenges attempts at classification [8].

[1] http://wiki.apache.org/
[2] http://wiki.java.net/
[3] http://esw.w3.org/topic/FrontPage

4.1. The features and assets of Platypus Wiki

The first thing to consider is the decision to represent every RDF resource in the same way as a Wiki page. A Wiki page always has a URL, while an RDF resource always has a URI. In a Semantic Wiki Wiki Web, this URI can also be reached with a URL, and sometimes URIs and URLs can be the same. While Wikis usually store information on a page in plain text, in Platypus Wiki, all pages are stored in a HTML file with metadata in RDF. The convention chosen to represent a link to a page is *namespace:pagename* which can be reached with URL http://hostname/namespace/pagename/.

If the user requests a URL http://hostname/namespace/pagename/index.rdf, the system returns only RDF metadata about the resources. Similarly if the user asks for http://hostname/namespace/pagename/index.html, we have chosen to return only the plain HTML content without any navigation bar, page header or footer. This enables content and metadata aggregation for other web sites. The idea is to enable metadata aggregation by specifying for every Wiki page where to find the RDF metadata (other than on the local host) to merge when users ask for the page. We have chosen to aggregate HTML content in the same way. It will be preferable to have a cache system to manage such aggregations, but this has not yet been implemented.

While relations between pages in a standard Wiki are HTML links, Platypus Wiki uses RDF properties between resources to construct "labeled HTML links". The information architecture in Wikis is an organic structure similar to a directed graph: a node is a Wiki page and a link represents a directed arc between two pages. In Platypus Wiki, RDF statements construct a directed labeled graph: a node is an RDF resource and a link is a RDF property. We have chosen to allow the user to select any RDF resource: subjects, objects or predicates. When the user clicks on a resource, it becomes the current resource and its RDF metadata is used to construct the navigation and presentation.

The layout of a page is based on three columns describing the current resource (see figure 1). The first column on the left is built with all the RDF statements that have the current resource as the object. Subjects of the same property are gathered in a box. The user can click on either the subject or the property, and this becomes the new current resource. In the second, central column, the main content of the page is presented. The statements that have the current resource as the subject and a literal as the object are shown at the bottom. The page content describes the concept represented by the URI of the current resource. The last column on the right is similar to that on the left, but contains all statements in which the current resource is the subject. This layout has been chosen, because in many lexicographic conventions, it is usual to write and read from left to right and from top to bottom. The RDF statements are thus as easy to read and understand as normal sentences.

Figure 1 - A screenshot of Platypus Wiki.

An essential feature of a Wiki Wiki Web is that

anyone can edit any page content. Links are generated from special conventions and the linking between pages establishes the structure of the navigation. In Platypus Wiki, users can edit the content in a similar way to a normal Wiki, but, in addition, they can also edit the RDF metadata. Metadata can be exploited by the user to build the navigation, or by programs for other purposes. Users are invited to add metadata themselves, and the system gives an immediate feedback in the navigation bars. The metadata can then be used by others. In addition, when users move form one page to another, they can always return to the previous resource by following the same path. If they click on a subject of a current resource, the subject becomes the current resource and the previous resource will be listed among the objects. The same thing happens with objects or any other resource. In other words, the system is like having bi-directional links.

Another important feature of Wikis is the automatic linking between pages. When users want to make a link to another Wiki page, they normally use a particular convention called WikiName or CamelWord. Some implementations use other conventions, like surrounding one or more words with square brackets. Platypus Wiki uses the namespace and pagename convention as *namespace:pagename*. As in standard Wikis, if the page exists and the user clicks on the link, that page becomes the current resource,

while if the page does not exist a form asking to create a new page is shown.

We have noticed that when users write Wiki pages, they frequently forget to use the convention to create links, because they are thinking about the content and do not know that other pages may or may not exist. To resolve this problem, Platypus Wiki offers the possibility to create "site links", "namespace links" and "page links". A "site link" consists of one or more words and an URL. It is like removing links from pages [9].

For example:

Word	**URL**
Java	http://java.sun.com/
Java Servlet	http://java.sun.com/products/servlet/
Platypus Wiki	http://platypuswiki.sourceforge.net/

Table 1 - Site links example

In each page, the Platypus Wiki engine replaces every word specified as a site link with the URL given. The engine is smart enough to replace the longest match first, so in the example given above, the word *Java* is replaced only if it is not followed by the word *Servlet*. Existing HTML links are maintained.

Therefore, users do not have to specify any links in a page and do not have to remember or to know of the existence of any page. They simply write the content, and when the page is published it will be automatically enriched with site links. If in a particular namespace or page, a word indicated as a site link must have a different URL, "namespace links" or "page links" can be specified. These links behave like site links, but their scope is limited to the current namespace or a particular page. Another possibility is to write standard HTML links, as they are always maintained.

One significant drawback of normal Wikis is their isolation from each other. Various solutions have been proposed by Wiki communities, e.g. InterWiki or SisterSites. Platypus Wiki could behave similarly using site links, but another way is by means of content and metadata aggregation. This is more flexible and makes it possible to interleave content and metadata from other Platypus Wikis. We have chosen to specify for each page a list of URLs from which to gather content and another list of URLs from which to gather metadata. These URLs can be local or remote, and it is irrelevant from where the information is aggregated. With a machine-machine or machine-user mechanism of subscription and notification, Platypus Wikis can be seen as a means to support a kind of *group forming network*.

In Platypus Wiki, we consider every resource, vocabulary item or concept of an ontology (class or property) in the same way on each page. Users can move between individuals or concepts following links built from metadata statements. In this way, users can learn the meaning of the vocabulary items and concepts used in the Wiki. Information, metadata, terms or ontology concepts are always managed in the same way. For a great number of intermediate users, it is important that this process is as simple as possible. When many users are able to edit and manage vocabularies or ontologies, a community can converge to create greater quality and produce better knowledge models. This means that ontologies can be constructed in a collaborative and simple way.

It is useful to monitor user activity continuously to allow the system to learn and react automatically so inducing metadata. Platypus Wiki counts each click on a link and uses this information to rank the results of queries or elements in lists displayed to users. We also gather explicit ratings for each Wiki page from users. Furthermore, we have found that it is important to track the user navigation paths. Monitoring which paths are followed most is useful in proposing navigational aids, winning user interest and correlating Wiki pages. Of course, this data can be used to generate RDF metadata automatically.

We think Semantic Wiki Wiki Webs should merge the world of the Wiki Wiki Web with that of the Semantic Web. In a Wiki Wiki Web, every page represents a concept, an object or an idea. From the point of view of Wikis, each page in Platypus Wiki is grouped under a "topic": the first level directory represents a set containing related pages. From the point of view of the Semantic Web, each page can be a class, a property or an individual. In the case of Platypus Wiki, the first level directory is the namespace of an ontology (a set of concepts and the relations between them) or a knowledge base (a set of individuals). This is an attempt to move towards the concept of *monocline grouping* explained by Alan Cooper [10] in his discussion of hierarchies as a non-natural concept for most people. The concept of monocline grouping is very similar to the XML namespace also used in the Semantic Web.

4.2. Implementation

Platypus Wiki is implemented in Java as a web application under Apache Tomcat as the servlet container, and using servlets and JavaServer Pages (JSP). Java offers various advantages in comparison to other languages. First of all, it can be used under

different operating systems. Another important consideration is the availability of one of the most advanced frameworks to build Semantic Web applications, including a rule-based inference engine, Jena[4]. Platypus Wiki uses Jena to manage RDF models and for a set of limited reasoning features. Jena is an open-source project, which means that its functionality can be analyzed in depth and modified as required.

Platypus Wiki has been implemented following the recommendations of the Model View Controller (MVC) paradigm. There is a servlet acting as a controller which intercepts all the HTTP requests. The controller checks if the request is a static or dynamic resource. In the first case, it forwards the request to the default servlet, while in the second case, an action is undertaken and the correct view is returned to the user. The views are built using JSP pages, and the model is represented by RDF metadata and HTML on each Wiki page. For simplicity, portability and performance, data is stored in the file system as plain text files. All Wiki pages are saved in a flat tree structure of only two levels. The name of a first level directory indicates a namespace, while the second level represents a term in a namespace. Each term and namespace is described in a Wiki page together with RDF metadata. Jakarta Lucene is used as a search engine for full-text and metadata searching

Platypus Wiki is an open-source project, and source files can be downloaded from http://platypuswiki.sourceforge.net/.

4.3. The uses of Platypus Wiki

Platypus Wiki can be used as a Personal Knowledge Management system and as a Wiki system. The aim is to manage a community of people, so features to aggregate information and metadata are fundamental. An organization can gather and merge information from its employees or share knowledge with other centers. Platypus Wiki is also valuable as a learning system, as using it to teach others is a good way to learn . William Glasser, an American psychiatric, said "we learn 95% of what we teach someone else".

In general, a Semantic Wiki Wiki Web can improve *lateral thinking* and association and help users to exchange useful information. Platypus Wiki allows users to create RDF classes, properties or individuals as a Wiki page. The quality of the data and metadata is given by the collaboration between the users. At present, Platypus Wiki does not enforce any constraints, so the resolution of any incoerencies is left to the communities.

5. Conclusion

In conclusion, we have proposed a meeting point between the Semantic Web and Wiki Wiki Webs, which we have called the Semantic Wiki Wiki Web. We have implemented this idea in Java and named our prototype Platypus Wiki. A Semantic Wiki Wiki Web can be used by a single person, a community or, better, by a Community of Practice to create and share information, vocabularies, ontologies or a generic knowledge base in a collaborative way.

Platypus Wiki is an ongoing project that started on 23th December 2003. The features described in this article are essential for a Semantic Wiki Wiki Web, but we are working to extend and to improve Platypus Wiki. In particular, we are trying to find a way to classify Wiki pages automatically into topics by using techniques from machine learning. Another important step is to identify the most common vocabularies and to implement features that add value to the end user. Possible vocabularies are, for instance, RDF Schema, OWL, Dublin Core and FOAF. An example would be to show the complete taxonomy of subclass relations when the current resource is an individual of rdf:Class.

We believe that it is important to give immediate feedback to the end user. Anytime an RDF statement is added to a Wiki page, users must have improved access to information. In this way, users are more encouraged to contribute by adding metadata. Moreover, use of an inference engine may be helpful in improving the quantity and quality of metadata. If a reasoning engine is used, the user adds RDF statements and the engine augments these statements by deducing others.

Platypus Wiki is built to be a Semantic Web platform with which to experiment and to try other ideas. It may be useful to extend Platypus Wiki towards a plug-in architecture. It would also be very helpful to study the appliance of trust and assess the reputation of users writing anything as an RDF statement.

We hope that the Semantic Wiki Wiki Web will be an important step in allowing users to participate in building the Semantic Web, and we envisage that Weblog, Wiki Wiki Web, P2P networks and collaborative annotation systems will converge to lead the World Wide Web towards a Semantic Web.

[4] http://jena.sourceforge.net/

10. References

[1] Bo Leuf, and Ward Cunningham, *The Wiki Way: Collaboration and Sharing on the Internet*, Addison-Wesley, 1st edition, April 2001.

[2] Ward Cunningham, *Portland pattern repository's wiki*, http://www.c2.com/cgi/wiki.

[3] Wikimedia Foundation, *Wikipedia: The Free Encyclopedia*, http://www.wikipedia.org/.

[4] Etienne Wenger, *Communities of Practice: Learning, Meaning, and Identity,* Cambridge University Press, 1st edition, December 1999.

[5] Etienne Wenger, Richard McDermott, and William M. Snyder, *Cultivating Communities of Practice*, Harvard Business School Press, 1st edition, March 2002.

[6] Italo Calvino, *Lezioni americane*, Mondadori, 14th edition, September 2000.

[7] Tim Berners-Lee and Mark Fischetti, *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by its Inventor*. Harper SanFrancisco, 1st edition, September 1999.

[8] Eco Umberto, *Kant e l'ornitorinco,* Bompiani, 1st edition, October 1997.

[9] Theodor Holm Nelson, *I don't buy in*, http://ted.hyperland.com/buyin.txt.

[10] Alan Cooper and Robert Reimann, *About Face 2.0: The Essentials of Interaction Design*, John Wiley and Sons, 2nd edition, March 2003.