

On Integrating a Semantic Wiki in a Knowledge Management System

Flavio De Paoli, and Marco Loregian
University of Milano-Bicocca (Italy)
{depaoli, loregian}@disco.unimib.it

Abstract—The use of knowledge management systems is often hampered by the heavy overload for publishing information. In particular, uploading a document and then profiling it with a set of meta-data and keywords is a tedious and time-consuming activity. Therefore, one of the main goals for such systems should be to make publishing of explicit knowledge as natural as possible. In the project described in this paper, we exploit a semantic wiki editor to support document publishing by means of textual descriptions augmented by ontology-defined annotations. Such annotations are then managed as entries in metadata profiles. Moreover, we can publish semantic-wiki-based documents that do not require any further activity to be profiled and included in a knowledge base as they are self-describing. The semantic wiki project is part of a collaborative knowledge management system that has been developed to support project teams and communities of interest.

I. INTRODUCTION

Workers in a modern organization need to organize, share, and retrieve knowledge that is collaboratively generated in heterogeneous workgroups. Various knowledge management (KM) systems have been proposed to support such fundamental tasks. However, experiences showed the limited efficacy of group tools and KM systems that prevented from a real adoption by their users. In particular, we are interested in two main socio-technical problems that can heavily invalidate the process of knowledge creation and circulation within organizations [1]: (i) the involvement of users in the collaborative process of publishing and editing documents, so that they can perceive it as a rewarding activity for their work and experience; (ii) the enhancement of the semantic capabilities of the ontology-based services provided by systems already in use.

In the MILK project (Multimedia Interaction for Learning and Knowing, [2]), a solution was developed to support the workers of an organization in exchanging knowledge by providing an environment to let them share ontologies, documents and documents' profiles. Moreover, users' and communities' profiles are supported and used to personalize the system behaviour. By using the MILK system, information navigation and retrieval can be accomplished according to the *context* in which it occurs, where context is defined as a situation composed of technological and social characteristics, and application objectives [3], [4].

Collaborative processes are enabled by effective profiling, which is in turn enabled by agreed and shared ontologies and by natural interaction means to exploit them. The issue

addressed in this paper is how to promote the involvement of people in creating and exchanging knowledge by providing them with an easy way of publishing group knowledge. The chosen approach is to combine the definition and sharing of personal ontologies with the creation of profiles and annotation of documents using a wiki interface [5]. A suite of tools that facilitate the definition and the evolution of personal ontologies, their use in common activities, such as Web browsing, and KM specific, such as document profiling, is described.

The rest of the paper is organized as follows. Section II outlines the MILK project, which is the environment that embed the proposed interaction techniques. Section III presents how ontologies are used in MILK. Section IV describes the exploitation of ontologies in creating MILK profiles, and semantic-wiki annotations. Section V presents related work. Finally, Section VI summarizes the paper contribution and outlines future work.

II. THE MILK SYSTEM

MILK is a KM system that allows a community of persons to store, organize and retrieve documents in a multimodal and multichannel way. The multi-tiered architecture of the system (Fig. 1, [2]–[4], [6]) includes three basic kinds of components: *interaction managers* that are devoted to the presentation and the interaction; *knowledge managers* whose tasks are defining, maintaining, indexing, and searching element profiles; and finally *archiving systems* — e.g., personal data management systems, and document management systems.

At KM level, MILK defines a profiling mechanism that unifies knowledge descriptions associated with documents,

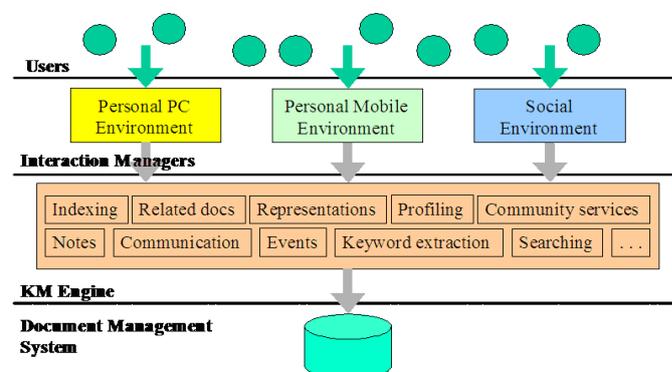


Fig. 1. The MILK architecture

people, projects and communities (collectively named *elements*). Any kind of element can be included in the same indexing and retrieval process. So, for example, given the profile of a project, it is possible to collect information about any other project that has similarities with the current one, know who is interested in the same subject, and identify documents regarding the same topics. The profiling process is driven by ontologies that describe specific domains [7]: ontologies can be used to edit elements' metadata, and are then exploited by MILK's Metadata Management System to compute correlations between elements — i.e., to identify related topics.

At interaction level, the aim of the system is to provide users with various contextualized interfaces and interaction mechanisms fitting different working situations, supplying users with different views over the same contents, to promote awareness and learning according to the activity the user is undertaking [8]. Moreover, user interfaces are requested to provide multimodal interactions according to different user terminals.

The need of supplying multimodal and multichannel presentation of documents — meaning, to be able to present the same document in different formats according to the most suitable device currently available to the user — led to an important choice in MILK's information architecture. MILK documents are collection of files, each one proposing a *representation* of the actual content of a document. For example, for each document might, or rather should, exist a representation that is well-suited for a regular PC (e.g., the full *pdf* version of a paper), one for large displays (e.g., a slide-show presenting the content), and one for handheld devices (e.g., the textual abstract of the paper). It is important to remark that all these files should be considered as being *the same* document or, at most, different *affordances* of the same document [9], [10]. Thanks to profiling, the MILK system is able to manage the different representations as “attachments” to the same element.

The use of profiles to support indexing and document retrieval has the additional advantage of being more accurate and less computationally expensive than dealing with textual, or multimedia documents.

The MILK system has been designed — and developed — to be accessible through three main channels: desktop PCs, handheld devices (e.g., mobile phones and PDAs), and large screens for synchronous collaborative interaction. The last two devices are mostly devoted to information access. So, for example, a large interactive screen can be used to support a face-to-face meeting by giving access to appropriate documents and people.

Desktop PCs have the crucial role of supporting most knowledge and information creation. In this paper we concentrate on this aspect: how to improve keyword-based profiling and annotation of documents using a PC GUI.

A. MILK desktop client

MILK on the desktop helps people to manage information and easily access the right sources for creating new con-

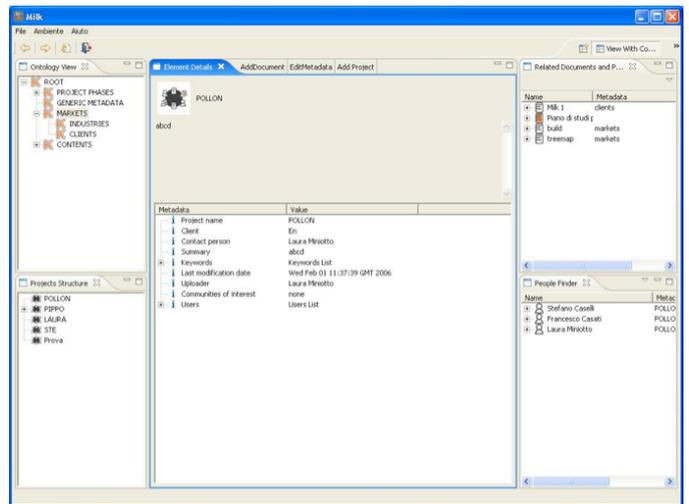


Fig. 2. The MILK desktop client GUI, showing the “View With Context” presentation.

tents [6]. In most KM systems, users are enabled to *search* repositories to get information; instead, MILK supports users to *discover* information while they are using the system. Discovery is supported by automatically computing and presenting information that are related to what the user is doing and displaying on the screen. Fig. 2 illustrates the concept in a prototype interface that displays a document surrounded by elements — more precisely, the element profiles — that are related to the content of that document. This interface style is named “view with context” to say that any displayed element is surrounded by its context [3]. For each document different views are available: in particular, a view presents the *detail* of the element, i.e., its full profile, while the *edit metadata* view allows to edit the fields composing the profile of the element.

Through the desktop client, MILK users can access documents in a shared repository and use a centralized ontology to profile the documents they want to share. Moreover, (i) thanks to the features described in the next sections, they can adopt personal (multiple [7]) ontologies with which they can create customized views over the repository; (ii) a space is assigned to each user for private work-in-progress documents — such workspace is called the Limbo project and can also contain sub-spaces (i.e., be structured with sub-folders). In this sense, each user is able to manage a personal knowledge base and to interact with a shared one according to customized, context-aware views. The repository is an FTP server that grants access according to policies that are defined at organization level and enforced by an LDAP server.

III. MILK PROFILING WITH ONTOLOGIES

In MILK, the task of publishing documents is basically supported in two ways: *automatic*, via FTP upload, and *user-driven*, through the MILK GUI (Fig. 2). In the former case, the MILK archiving server can be mounted as an external drive for the file system (as it can be easily done with the most common operating systems) and documents can be dropped in a project

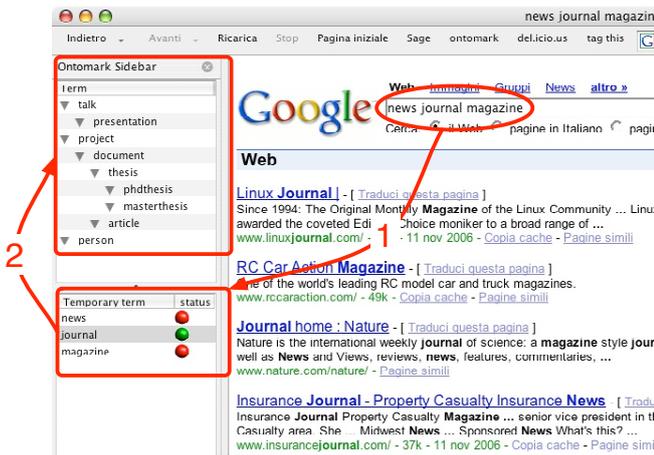


Fig. 4. The Mozilla Firefox extension, and the main interaction steps to define a personal ontology by adding a new term.

shared ontologies and repositories to include the terms used during browsing as personal ontology terms. We are testing an extension for Mozilla Firefox that monitors the queries sent to Google (Fig. 4).

The user interface of the browser extension consists of a sidebar with an upper “permanent” panel, and a lower “temporary” panel. Whenever a Google page is loaded, the keywords used in the search are extracted by parsing the page URL and added to the bottom panel of the sidebar (step 1 in Fig. 4). The list of keywords in the panel is temporary, and it is reset every time Firefox is started. By right-clicking on any of the keywords currently displayed, the user can search the term again with Google (like with a bookmark) or make the keyword part of a permanent set (step 2 in Fig. 4).

When the choice to make a keyword part of the permanent personal ontology is made, a query can be sent to Swoogle³, via its Web Service APIs, to find out if the term is present in other existing ontologies. To see if matching terms exist, a new browser tab with the results of the query is presented to the user to let her select the most appropriate term and solve possible ambiguities. In practice, the user is requested to associate the term with an existing RDF namespace — picking among those returned by Swoogle — or to use a default one.

At the moment, direct manipulation of the set of terms stored in the permanent set — beyond the re-arrangement of the terms according to the *is-A* relation, which can be done with the extension — exploits external editors, such as Protégé⁴.

Besides using terms as bookmarks for Google searches, the general utility of creating a personal ontology is to use it in KM systems such as MILK (i.e., by MILK GUI) to better organize personal knowledge or retrieve information from a shared repository: needless to say, it is the vision of the Semantic Web [12].

³<http://swoogle.umbc.edu/>

⁴<http://protege.stanford.edu/>

For this reason we decided to be compliant with Semantic Web technologies: the upper panel relies on a RDF file that permanently stores a user’s personal ontology, that can be loaded and managed by the MILK client GUI (as described next) as well as by any other RDF-based application.

C. Ontologies in the MILK client

The ontology panel in the MILK desktop GUI (Fig. 2) is composed of multiple tabs: besides the one for the shared reference ontology — that is retrieved from the MILK server, — others can be added to present the candidate terms and personal ontologies (as the ones described by Boselli and De Paoli [7]). Such ontologies are RDF files that can be loaded from the local hard drive or from the Web, via URLs. As described already in Section II-A, personal ontologies are part of the process that leads to the creation of the View with Context for each user and play fundamental role in the publishing of novel knowledge in the working community.

When a document is profiled with terms that are only part of a personal ontology such terms are managed as new candidate terms with the advantage of carrying along additional information. That is, a portion of the personal ontology, rather than a simple keyword, is sent to MILK’s ontology management component: ancestors (according to standard *is-a* or *part-of* relations), descriptions and relations are used to qualify the terms whenever the users correctly specify the namespace of the ontology from which they took the term as a reference, or when users are able to maintain their own ontologies, e.g., using Protégé.

To take advantage of this feature, MILK administrators could exploit ontology merging techniques [3] to build a unified view of the ontology — over approved and candidate terms, exploiting additional information acquired from personal ontologies, — so to have a semi-automatic insertion of the candidate terms.

Further usage of ontologies, granting an ever higher expressive power, is achievable through semantic annotation, as described in the next section.

IV. MILK SEMANTIC WIKI

The MILK GUI is a standalone application, built on Eclipse Rich Client Platform⁵. The advantage of this choice is to deliver a (cross-platform) tool that is easily extensible, via plug-ins, to allow for rapid proofing of new ideas.

The interface is structured in *perspectives* that collect related *views*. For example, the View with Context perspective is composed of a main view, which fills most of the available space, surrounded by smaller panels displaying ontologies, related elements, project structure, and available people (Fig 2). The main view (i.e., the edit metadata view of Section II-A) is essentially form-based, meaning that metadata is entered in profiles by filling in the fields of a form. This is widely recognized in Computer Supported Cooperative Work (CSCW) literature as a tedious task. In particular, this kind of interaction

⁵<http://www.eclipse.org/rcp/>

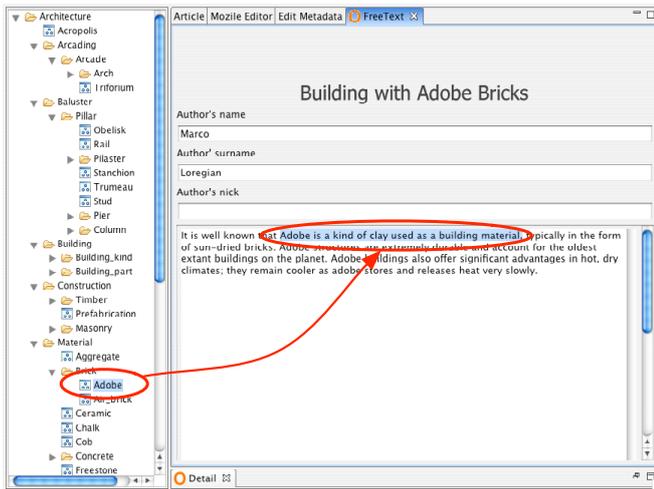


Fig. 5. A detail of (an early version of) the Wiki perspective embedded in the MILK client.

is not perceived as a collaborative activity. Forms suggest a rigid conception of profiles, whereas the MILK Metadata Management System was designed for more flexibility (e.g., to deal with heterogeneous elements).

The popularity of tools such as Wikipedia [13] shows that wiki-style interaction could be the key factor to allow users to generate new contents encompassing various and novel representations (Section II) by completing existing element profiles with information that can be easily edited (with elementary wiki syntax) and exploited, for example, to build richer views on the knowledge base.

For these reasons we are testing a “Semantic Wiki perspective” in the MILK client (Fig. 5). It is a wiki in the sense that users can annotate elements’ profiles as free-text, and we claim it to be a *semantic* wiki because of the enhancement it carries in knowledge processing, being tightly integrated with MILK’s ontologies [14], [15]. The definition of a wiki representation enhance the content management capabilities of the system, for example to allow for more precise computation of the view with context, taking annotations into account. Moreover, the content created using the wiki can also complement existing documents with lightweight representations to be used when a specific representation is not available. For example, if a user wants to interact with a document on a large screen but there is no slide-show to be presented, the wiki description of the document can be used (by means of XSLT and CSS) to render a compact display of the element. Similarly, the wiki description could be easily displayed on handheld devices.

In this sense, the semantic wiki perspective plays the important role of providing direct access to the creation of machine-processable documents, at the same time facilitating the human-computer interaction. In the current prototype, the only actions to be performed on a form are the assignment of a title, and the choice of the project or community of which the document will be part (otherwise the file cannot be correctly created on the archiving server). Then, an editor

like the one outlined in Fig. 5 opens and the user can start typing the content.

As already described, MILK ontologies are presented in the panel on the left, and annotations can be made on the document by simply selecting words or sentences in the text area and then dragging one or more ontology terms on them. By this action, a RDFa⁶ annotation is created in the document, and the user perceives it through the change of the annotated portion of the text that turns underlined. Code-level editing features are also available, so that annotations can be added and removed directly in the source code. If the term is taken from the personal ontology, but it is not part of the shared MILK ontology, then the term is added to the list of candidates (with all its properties and relations) when the document is saved.

A. Scenario

An example clarifies the use of the semantic wiki to support collaborative ontology evolution; it refers to annotations in the field of architecture [16]. The scenario involves an architect that is temporary collaborating with a design firm. She is uploading a document explaining how *adobe* is used as building material. One of the first sentences of the document explains that *adobe* is “a kind of clay used as a building material, typically in the form of sun-dried bricks” (from the Oxford dictionary); therefore the architect creates an annotation by highlighting the sentence, clicking on *adobe* in her personal ontology and then dropping the keyword on the definition. As explained above, an annotation is created and *adobe* becomes a candidate for the main ontology. But what if the organization is already using the term “adobe” to identify the suite of programs of the well-known software company? The MILK ontology administrators can recognize the conflict and solve it thanks to the additional information provided along with the candidate term (i.e., the personal ontology fragment and the annotation).

Summarizing, users can edit profiles by adding annotations through the MILK Graphical User Interface (GUI) in three ways, in increasing order of carried information:

- 1) directly type a new term in the proper form field. In this way no additional semantic information (i.e., definitions, properties, and relations with other terms) can be provided. In this case, the keyword is just a candidate term;
- 2) drag and drop a term from a personal ontology. In this case, an ontology fragment, rather than just a term, is proposed as a candidate;
- 3) use terms from the personal ontology in the Semantic Wiki perspective, to create document annotations.

B. Some Wiki details

MILK’s wiki was designed according to a simple, yet consolidated architecture [15] (Fig. 6). The user interface has been implemented with Eclipse RCP platform, and four main views are currently available:

⁶<http://www.w3.org/TR/xhtml1-rdfa-primer/>

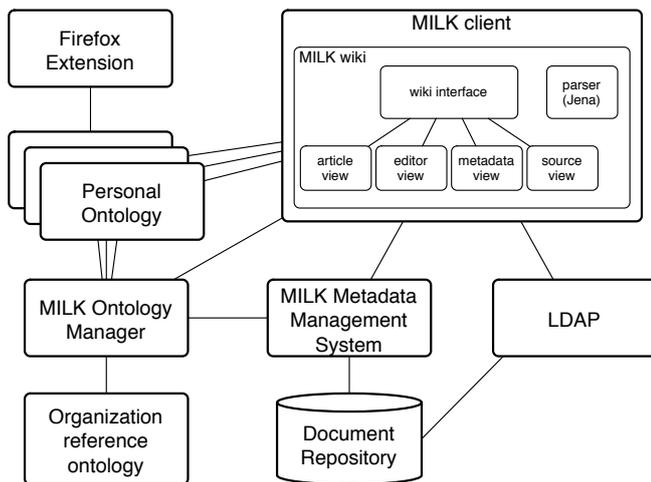


Fig. 6. The architecture of the system: only elements relevant to the wiki perspective are shown.

- the *article* view is a read-only display of the wiki representation of the document. With this view users can have a general view over the annotations of the document: annotations are outlined by the underlined text. Using mouse tooltips it is possible to view annotation details. Using XSLT and CSS technologies, the wiki description can be rendered on a range of devices or exported in various formats, with various appearances;
- the *editor* view presents a compact editable view of the document: the text and the annotation can be modified by picking terms from the available ontologies. Also in this view, annotations are only shown by text underlinings;
- the *edit metadata* view presents the list of the annotations in a form: metadata can thus be quickly edited and the changes are automatically inserted in the wiki document;
- the *source* view allows the user to directly edit the source code of the wiki document (i.e., RDFa annotations): tags are shown and can be directly edited (Fig. 5) in the body of the document.

To make changes permanent, the document has to be saved: upon this action, the MMS is in charge of parsing the wiki document (the wiki perspective relies on Jena APIs to generate documents) to extract information and update the document profile. The RDFa document is stored in the repository along with other representations and documents.

Current implementation does not allow users to write or perform queries on the document repository directly from the wiki, for example using SparQL⁷ or XQuery⁸: the reasons are due to MILK's *knowledge discovery* approach rather than to technical issues — the design or integration of a query engine able to translate the queries for the MMS could be very useful and is planned as future work.

⁷<http://www.w3.org/TR/rdf-sparql-query/>

⁸<http://www.w3.org/TR/xquery/>

MILK supports the definition of permissions and access rights and policies so to separate the creation and editing phases of a document from the browsing of the knowledge base. For example, only certain views are displayed when browsing documents, and the switch to “edit mode” requires an explicit action by the user (i.e., to click a button or open a different perspective). Thanks to such separation it is also possible to deploy different versions of the MILK PC client — for example, a Java Web Start browse-only version for users working from casual locations or shared workstations.

V. RELATED WORKS

The work presented in this paper took inspiration from other efforts made in developing semantic wikis. Among others, SEMPERWIKI, developed by Oren et al. [14], [15], [17], [18], and IkeWiki [19], [20]. While other wikis have been designed as standalone tools, with the purpose of creating an homogeneous, yet processable knowledge base, the project described here aims at going a step further. With respect to KM, it seems rather unrealistic to consider wikis apart from other existing and already adopted collaborative tools, or as a new technology to be applied only when starting to build new collaborative knowledge bases. Thanks to the simplicity with which annotation can be added to existing documents, wikis should instead be integrated with technologies in use. In this paper we presented an example of how this can be done with MILK, which relies on a file system managed by an FTP server. An advantage of this technique is that the integration can be totally transparent for end users, that could also decide to ignore the feature. Other integration experiments could be carried on with respect to other existing document management platforms, such as BSCW (formerly adopted in MILK) [2].

Moreover, thanks to the reuse of semantic terminologies — enabled by MILK client ability to load ontologies from various locations and by the Firefox extension to edit personal ontologies — users can be supported all across their main working and learning activities without making them quit the tools to which they are used.

The browser-extension approach is already widely in use to enrich the browsing experience with additional information. For example, the Sage extension⁹ makes an RSS/Atom feed reader available inside the browser as a sidebar, reducing the need to use an external application (i.e., feed aggregator) to automatically collect information about topics interesting to the user.

From a cognitive and interactive perspective, the process of creating and editing a personal ontology with the extension presented in Section III-B is similar to the organization of regular Web bookmarks [21], [22]. Such an activity is often considered unnecessary as long as the number of bookmarks is limited, and all the favorite URLs can be kept together unsorted, but most Web users tends to develop personal bookmark hierarchies (with folders and subfolders by topic)

⁹<http://sage.mozdev.org/>

to sort their references. The result of such operation, which usually takes only a few minutes, is an explicit representation of the interest of a person. Additionally, our browser extension can be used to express such conceptualization in a standard language, thus, ready to be shared with other users. It is not just *social bookmarking* or annotation [23], [24], in that the generated ontology remains personal until parts of it are willingly used in a collaborative setting and, thus, shared.

Other interesting research issues may arise from investigating MILK's interoperability with other tools for the Semantic Web. For example, the knowledge base generated by MILK users could be accessed by tools such as DBin [25], [26] to enact peer-to-peer sharing of documents' profiles. Being file URIs part of the profile, and being files published and accessible via FTP (to authorized persons, where a central directory can be used to define policies and workgroups), complete personal installations of the system (featuring server and client components) could easily be deployed. Early, autonomous, experiments have already been made in this direction [4].

VI. CONCLUSION AND FUTURE WORK

In this paper we have presented the ongoing work to design a KM tool integrated with the Semantic Web philosophy [7]. We have outlined how a semantic wiki can be introduced and exploited in a knowledge management system to effectively support explicit knowledge creation. The advantages of adopting such an approach is to promote the publishing activity by letting users write textual descriptions augmented with semantic annotations, and to promote collaborative profiling.

We believe that users will be more involved in the cooperative activities since they can add personal annotations and use their personal ontologies. Moreover, the possibility of discovering new terms and relations among documents is a rewarding feature to convince persons to collaborate.

Semantic interoperability (between users and applications) is facilitated by (i) the capability of the Firefox Extension to preserve references to standard namespaces (Section III-B), (ii) MILK features for co-occurrence analysis [11], and (iii) MILK administrator interfaces for browsing candidates and their properties and for enacting merging algorithms [4].

Apart from the desired features mentioned along this paper, future work is the finalization and evaluation of the implementation, which is currently in a preliminary version that cannot be distributed to our test users. Other future works include the integration of email and RSS feeds (by means of a Mozilla Thunderbird extension or of a Chandler parcel) with MILK so that users can edit their ontologies and possibly also publish documents directly from the mail client.

ACKNOWLEDGEMENT

This work was partially supported by the MILK project (IST-2001-33165): <http://www.cootech.disco.unimib.it/milk/> Some of the features presented here as part of MILK have been introduced as results of the previous research in the ATELIER project (IST-2001-33064): <http://atelier.k3.mah.se/>

- [1] I. Nonaka and H. Takeuchi, *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press, 1995.
- [2] A. Agostini, S. Albolino, F. De Paoli, A. Grasso, and E. Hinrichs, "Supporting communities by providing multiple views," in *Proceedings of the Second Communities and Technologies Conference 2005*. Springer, 2005, pp. 437–456.
- [3] F. De Paoli and M. Loregian, "Context-aware Applications with Distributed Ontologies," in *Advanced Information Systems Engineering, 18th International Conference, CAISE 2006, Luxembourg, June 5-9, 2006. Proceedings of Workshops and Doctoral Consortiums*, T. Latour and M. Petit, Eds. Presses Universitaires de Namur, 2006, pp. 869–883.
- [4] —, "Enhancing Semantic Interoperability in Collaborative Systems," in *Proceedings of the Eighteenth International Conference on Software Engineering and Knowledge Engineering (SEKE 2006)*. Knowledge System Institute Graduate School, 2006, pp. 29–34.
- [5] B. Leuf and W. Cunningham, *The Wiki Way: Collaboration and Sharing on the Internet*. Addison-Wesley Professional, April 2001. [Online]. Available: <http://www.amazon.com/exec/obidos/redirect?tag=citeulike-20&path=ASIN/020171499X>
- [6] A. Agostini, S. Albolino, R. Boselli, G. De Michelis, F. De Paoli, and R. Dondi, "Stimulating knowledge discovery and sharing," in *GROUP'03*. ACM Press, 2003, pp. 248–257.
- [7] R. Boselli and F. De Paoli, "Semantic Navigation through Multiple Topic Ontologies," in *Proceedings of SWAP 2005, the 2nd Italian Semantic Web Workshop, Trento, Italy, December 14-16, 2005*. CEUR Workshop Proceedings, 2005, vol. 166. [Online]. Available: <http://ceur-ws.org/Vol-166/7.pdf>
- [8] T. Rodden, "Populating the application: a model of awareness for cooperative applications," in *CSCW '96: Proceedings of the 1996 ACM conference on Computer supported cooperative work*. New York, NY, USA: ACM Press, 1996, pp. 87–96.
- [9] D. A. Norman, *The Design of Everyday Things*. Doubleday, Ney York, 1990.
- [10] —, *The Invisible Computer*. Cambridge, Massachusetts; London, England: The MIT Press, 1998.
- [11] S. Calegari and M. Loregian, "Using Dynamic Fuzzy Ontologies to Understand Creative Environments," in *Flexible Query Answering Systems 2006, 7th International Conference, FQAS 2006, Milan, Italy, June 7-10, 2006*, ser. Lecture Notes in Computer Science, H. L. Larsen, G. Pasi, D. Ortiz-Arroyo, T. Andreasen, and H. Christiansen, Eds., vol. 4027. Springer-Verlag, 2006, pp. 404–415.
- [12] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, vol. 284, no. 5, pp. 34–43, May 2001. [Online]. Available: <http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21>
- [13] J. Giles, "Internet encyclopedias go head to head," *Nature*, vol. 438, pp. 900–901, 2005. [Online]. Available: <http://www.nature.com/news/2005/051212/full/438900a.html>
- [14] E. Oren, M. Völkel, J. G. Breslin, and S. Decker, "Semantic wikis for personal knowledge management," in *DEXA*, ser. Lecture Notes in Computer Science, S. Bressan, J. Küng, and R. Wagner, Eds., vol. 4080. Springer, 2006, pp. 509–518.
- [15] E. Oren, J. G. Breslin, and S. Decker, "How semantics make better wikis," in *WWW '06: Proceedings of the 15th international conference on World Wide Web*. New York, NY, USA: ACM Press, 2006, pp. 1071–1072.
- [16] M. Loregian and M. Telaro, "Dynamic Ontologies and Cooperative Learning," in *Supplements to Proceedings of COOP 2004*, 2004, pp. 73–80. [Online]. Available: <http://klee.cootech.disco.unimib.it/Atelier/lore gian-telaro.pdf>
- [17] E. Oren, R. Delbru, K. Möller, M. Völkel, and S. Handschuh, "Annotation and navigation in semantic wikis," in *SemWiki (ESWC)*, June 2006.
- [18] M. Völkel and E. Oren, "Towards a Wiki Interchange Format (WIF) – opening semantic wiki content and metadata," in *SemWiki (ESWC)*, June 2006.
- [19] S. Schaffert, "IkeWiki: A Semantic Wiki for Collaborative Knowledge Management," in *1st International Workshop on Semantic Technologies in Collaborative Applications STICA 06, Manchester, UK*, 2006.

- [20] S. Schaffert, A. Gruber, and R. Westenthaler, "A semantic wiki for collaborative knowledge formation," in *Proceedings of SEMANTICS 2005 Conference.*, Vienna, Austria, 2005.
- [21] D. Abrams and R. Baecker, "How people use www bookmarks," in *CHI '97: CHI '97 extended abstracts on Human factors in computing systems.* New York, NY, USA: ACM Press, 1997, pp. 341–342.
- [22] D. Abrams, R. Baecker, and M. Chignell, "Information archiving with bookmarks: personal web space construction and organization," in *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems.* New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1998, pp. 41–48.
- [23] D. R. Millen, J. Feinberg, and B. Kerr, "Dogear: Social bookmarking in the enterprise," in *CHI '06.* New York, NY, USA: ACM Press, 2006, pp. 111–120.
- [24] X. Wu, L. Zhang, and Y. Yu, "Exploring social annotations for the semantic web," in *WWW '06.* New York, NY, USA: ACM Press, 2006, pp. 417–426.
- [25] G. Tummarello, C. Morbidoni, F. Piazza, P. Puliti, F. Saletti, and J. Petersson, "P2p multimedia annotation and browsing based on semantic web and mpeg-7: An overview of the dbin project," in *Fourth MUSICNETWORK Open Workshop: Integration of Music in Multimedia Applications*, Barcelona, September 2004.
- [26] G. Tummarello, C. Morbidoni, P. Puliti, and F. Piazza, "The dbin semantic web platform: An overview," in *Workshop on The Semantic Computing Initiative (SeC 2005)*, Chiba, Japan, May 2005. [Online]. Available: <http://www.instsec.org/2005ws/papers/tummarello.pdf>