A Social Network Ontology for Semantic Web-Enabled Collaboration

Thomas K. Yan and H. Peter Dommel
Santa Clara University, Computer Engineering
500 El Camino Real, Santa Clara, CA 95053, USA
{thomaskyan@gmail.com, hpdommel@gmail.com}

Abstract - Social networks have first been studied by social scientists to understand the patterns of relationships between people. More recently, social networking on the Internet capitalizes on the Web's latent structure as a meta-network of social connections to boost computer-supported collaboration in conjunction with the use of Semantic Web metadata. Such metadata are generally machine-created and per se lack certification or trust grading. This paper examines how social network information can be used to accelerate reliable metadata creation. In particular, we focus on the application of Web annotations to leverage asynchronous collaboration through the Semantic Web by introducing the notion of a user's Web personality to model the various trust levels in collaboration.

Keywords: Semantic web, collaboration, ontology management, trust networks, user modeling.

1. Introduction

Much of the Semantic Web functionality envisioned by Tim Berners-Lee relies on ontologies. Creating ontologies is difficult, time-consuming, and expensive, reminding of the labor of knowledge engineering in expert system design, in particular if ontologies are designed to support automated inference envisioned by advanced Semantic Web applications [1]. Semantic Web information, which is a largely machine-readable extension to the existing syntax-driven Web, allows applications to uniformly take into account semantic metadata of Web documents. The complexities of ontology development have given rise to an emerging discipline called ontological engineering.

Social networks illustrate relationships between two people in a graph to derive significant paradigmatic properties. Traditionally, such networks were used to better understand sociological behavior, but famous notions such as the "six degrees of separation" or the "small world phenomenon" can easily be generalized to both natural and artificial networks. Social networks can be viewed as either aggregate or egocentric. ContactMap [2] exemplifies an egocentric network type, where each user has a personal social network perspective. An aggregate social network is a single social network capturing the relationships of an entire group. Users sometimes find the use of aggregate social networks problematic, as they have a better understanding of their own social networks. Explicitly constructed social networks, where users manually add links to people they are connected to, have become popular on the Web. Data mining can also be used to construct social networks based on collaborative filtering and recommendations [3] or email interactions [2].

Collaborative applications have evolved tremendously in recent years and infuse the Web with novel ways such as wikis, blogs, content syndication, or tagging to help users to interact, learn from each other, and co-create. For example, TrustMail [4] is a collaboratively constructed social network using Semantic Web mechanics for spam filtering. The Friend of a Friend (FOAF) [5] ontology is widely used for constructing machine-readable homepages using social networking metadata, which, unlike traditional Web pages, can be combined with other FOAF documents to create a unified database. The marriage of social networks and the Semantic Web provides hence fertile ground to leverage collaborative practices among users. The primary challenge lies hence in the generation, introduction, and validation of Semantic Web information, keeping in mind applications, social relations, and communities of practice [6].

In this paper, we describe a novel collaboration architecture combining the power of social networks, the Semantic Web, and asynchronous collaboration. In order for any collaborative application to succeed, those using the system must benefit directly in its use. To this end, we use ideas from shared bookmarks, an increasingly popular Web application, and link bookmarks with annotations. By using collaborative Web annotation as an application, we illustrate how the Semantic Web can be used to enhance computer-supported collaboration through personal, trusted ontologies.

The rest of this paper is organized as follows: Section 2 discusses social networking at the interface with the Semantic Web and describes current annotation technology. Section 3 describes an ontology for creating a Semantic Web-based social network with a social notion of trust. Section 4 discusses how a collaborative Web annotation system can be augmented with social networks and the Semantic Web, with special focus on implementation issues. Section 5 concludes the paper.

2. Semantic Web Collaboration and Annotation

Shared Web annotations leverage the process of adding machine-readable metadata to Web content for automatic processing and inference. While annotation typically focuses on representing the semantics of Web documents for agent use, we focus on Semantic Web annotations in a collaborative system. Application developers can create a social network ontology that fits the needs of a specific application. However, the use of the Semantic Web eases the exchange of data between applications using different ontologies if the need arises. In contrast, as evidenced by businesses that exchange data using the Electronic Data Interchange approach, such isolated technologies do not solve data integration with different applications [7].

If different applications use the Semantic Web, even when using different ontologies, data can be exchanged if a minimum set of elements for global exchange of data within a domain existed [8], although representing knowledge in an ontology is itself often a tremendous effort and methods for creating mappings between ontologies are still an active area of research [9]. Some possible methods to provide a semantic mapping between ontologies are through negotiating agents, machine learning, or linguistic analysis. The Semantic Web also allows developers to take advantage of previous work through the reuse of ontologies.

Objects in the Semantic Web are represented using a Uniform Resource Identifier (URI). Suppose that a user finds the URI of another user in a Semantic Web social network. Since the URI represents the person in the entire Semantic Web, that same URI can be used to answer queries unambiguously assuming the existence of a Semantic Web search engine such as Swoogle [10].

In terms of implementation, Semantic Web technologies involve many standards and recommendations including the Resource Description Framework (RDF), RDF Schema (RDFS), the Web Ontology Language (OWL), various query languages, and inferencing of RDFS and OWL [7]. RDFS is flexible enough to build ontologies to represent people and the relationships between them, as successfully explored with the FOAF ontology. The flexibility of RDF allows specific applications to extend existing ontologies for their own needs. For example, TrustMail extends the FOAF ontology to include trust information in its representation of relationships between people. The OWL provides features not available in RDFS for applications that need a more powerful ontology language. Query languages such as the RDF Query Language (RQL) and the RDF Data Query Language (RDQL) understand the semantics of RDF vocabularies and can be used to query RDF documents.

These technologies, or a subset of them, are an essential part of any Semantic Web application. Therefore, Semantic Web application developers should be able to work with a high level API that implements these technologies, instead of having to implement them from scratch. Jena [11] is a Semantic Web toolkit that provides APIs for its standards and recommendations in the Java programming language. It is one of the most widely used Semantic Web programming toolkits and provides an RDF Java API that supports the creation, manipulation, and querying of RDF graphs. Jena supports the persistent storage of RDF in a relational database by storing RDF triples in a general-purpose table or property table for specific properties

Various systems have been introduced recently to create private and shared Web annotations, for example Annotea [12], where the flexibility of RDF allows users to create new subclasses of the "annotation" type to fit the needs of their applications, re-using elements from the Dublin Core metadata standard [13]. Annotea allows metadata to be stored on local machines, on a server, or in Web documents [14]. These are referred to as *data stores*, and users are required to know how to subscribe to data stores containing Annotea metadata. Haystack [15] aggregates all of a user's information from email, documents, calendars, and Web pages into a unified RDF metadata repository for centralized use. In contrast to Annotea, Haystack can process natural language annotations [16]. In another approach to Semantic Web annotation, blog entries are semantically enhanced with RDF metadata [17]. Blogs typically use a Really Simple Syndication (RSS) to create tables of content. Ontologies can be used to extend RSS in order to bring out the hidden semantics of blogs such as annotations and message chains.

3. Web Personality-based Ontologies and Trust

Semantic Web applications whose use extends beyond a single community must also deal with the issues of authority and trust. Since anyone can create metadata, establishing trust becomes an important issue as knowledge representations may be open to subjective interpretation and human-generated metadata may be plagued by deceit and ignorance on the part of the metadata creators.

In order to use social networks for collaboration with the Semantic Web, we must first define an ontology for social networks that includes a social notion of trust. The prevalent notion of digital trust is based on authentication; however, even if an information source sufficiently identifies itself, its information may be unreliable. Collaborative applications in particular can benefit from a social notion of trust, dealing with reputation, authenticity, and information quality.

Computing social notions of trust involves social network analysis. Algorithms similar to those used by search engines to rank the importance of Web pages can use the link structure of social networks to assign universal trust values to producers of information on the Semantic Web. Alternatively, with input from users who rate the users they are connected to, algorithms that gather information from the resulting social network are able to provide personalized trust values for each individual. Similarly, the FOAF ontology is used to create a social network, and FOAF can be extended to include user-specified trust values [4].

Golbeck *et al*. [18] measure trust on a numeric scale of 1-9, with 1 indicating that the user distrusts another user absolutely and 9 indicating that the user trusts another user absolutely. Users are able to specify a general trust rating for another user as well as trust values for specific topics. There are no restrictions on the topics that trust values can be specified for. By allowing users to specify trust values for other users, the ontology describes both trust and distrust among users. Bidirectional relationships in a social network may have a different trust value in each direction, reflecting user opinions in assigning trust, which may reflect trust in reality more accurately than trust automatically inferred from network proximity. Trust values for users or for specific topics allow applications to use those values that best fits its requirements.

However, it is possible that various ontologies describe the same topic, each tailored to a community of practice. Hence, numerous resources from different ontologies can be used for the same topic. If the user is concerned that the trust value specified for the topic is ubiquitous for the topic and not just for a specific ontology, then the user must specify the same trust values for resources from other ontologies representing the same topic or rely on sophisticated semantic mapping tools across different ontologies. This issue is more problematic in a universal social network than in one used within a single community of practice.

One possible solution is to differentiate between the notions of trust related to subjects and notions of trust related to a user's *Web personality*. Here we consider subjects to be matters where trust is based on knowledge, e.g., in computer repair. Despite the difficulties stemming from the flexibility of allowing resources from any ontology to be specified as subjects, this appears to be the best approach for trust in subjects. If a social network ontology is used within a community of practice, the need for semantic mappings between ontologies may not be a problem. It may be best to leave it to the users to decide how to deal with the complexity involved with it.

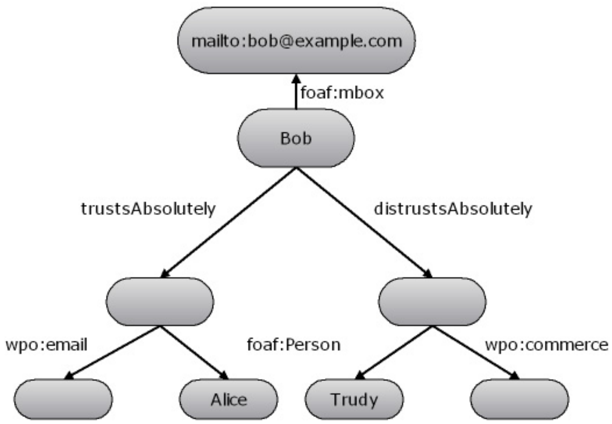

Figure 1. Logical diagram of an RDF model using FOAF and the Web personalities ontology.

We take a different approach with notions of trust related to a user's Web personality as a universal method for foot-printing user characteristics. In this paper, Web personality is defined by the user's behavior on the Web. Is a user more likely to send junk mail? Is a user more likely to lie about the conditions of products he or she is selling online? Since the areas of Web personality are broader than subjects, we propose an ontology representing Web personalities for universal use. This ontology is tied to the social network ontology by restricting trust values for Web personalities to use the vocabulary defined in the corresponding ontology, as depicted in Figure 1. Some sample classes of Web personalities are listed below.

Personality: a super class describing Web personality in general.
Commerce: a subclass of *Personality* representing how much a user is trusted as a buyer or seller in online commerce.
Email: a subclass of *Personality* representing how much a user is trusted to send emails that are worth reading.
Comment: a subclass of *Personality* representing a user's etiquette in message boards, chat rooms, or blog comments.
Collaboration: a subclass of *Personality* representing a user's reliability regarding collaborative activities on the Web.

Our Web personalities ontology uses the same nine levels of trust described above that ranges from distrusts absolutely to trusts absolutely. The ontology itself is simple, consisting of a single super class representing a user's Web personality in general and several subclasses representing general areas of Web usage where trust values can be applied.

An algorithm for computing trust values of other users for both subjects and Web personality in a social network is needed. Let t_{jk} represents the trust value between a source node *j* and a sink node *k*. If there is a directed edge in the social network from *j* to *k*, the trust value is the value the user represented by node *j* specified. If there is no directed edge, trust can be inferred as a weighted average of the trust values for *k* returned by each of *j*'s neighbors [4]. Greater weight is placed on the nodes closer to node *j*. This reflects the fact that people tend to trust information from sources that they trust and distrust information from sources they know they distrust. If the application uses specific trust values in the social network, the algorithm must take nodes with missing trust information into account. Applications can automatically assign trust values to more general trust categories if a user only specifies values for its subclasses. This will be helpful for applications that first try to find a specific trust value for a user and defaults to the least general super class if the value does not exist. One possible way to achieve this is to assign the average of the trust values of the subclasses as the trust value of a super class that was not explicitly set by the user.

When the algorithm encounters a node lacking a trust value for the desired class, the trust value of the least general super class of the target class or the general trust value is taken heuristically. Applications using such a social network can help assure that trust values are assigned to general super classes if users do not directly specify them. For example, with our Web personalities ontology, if a user assigning trust values for another user only specifies values for the *Commerce* and *Email* classes, the application can automatically assign the average of the two values to the *Personality* super class.

Disconnected users in the social network are problematic when trust values need to be computed. Disconnected nodes include nodes that may be part of a group but are not connected to the node that is requesting a trust value for it. The easiest solution is to not calculate a trust value for disconnected users since it is impossible to calculate a personalized trust value for them. Alternatively, if a Semantic Web search engine could be used to find nodes that are directly connected to the disconnected node, a trust value may be computed using those nodes.

A study of FOAF documents on the Semantic Web [19] indicates that disconnected users would be a common case, at least initially. At this time, FOAF is used for people to connect themselves with their friends. The study of FOAF documents from non-blog sites showed that most FOAF documents only provide information about the user and that groups of connected FOAF documents were typically small, with few groups having more than one in-degrees or out-degrees. It is projected that the topology of FOAF documents will evolve into larger connected structures with star-shaped patterns [19].

Applications using social networks must also facilitate adding Web personality trust values through an intuitive user interface. For example, online commerce sites typically allow buyers to assign a rating to sellers and vice versa. Instead of forcing the user to choose the Web personality class to assign the trust value to, the application can allow the user to set the rating and automatically assign it to the *Commerce* class.

Integrating social network power with Web personality ontologies has several benefits for users. Contributing to the goal of better data integration using the Semantic Web, applications using the same social network can share data by allowing positive trust values to be carried across applications. For example, a user with a high *Comment* trust value gained from proper etiquette in postings in a message board would be able to carry over the high trust value if he/she starts posting on another message board. The use of personalized trust ratings allows the existing members of the second message board to quickly assign their own opinions regarding a new user. In addition, assigning trust or quality values to Web content may leverage search algorithms by taking into account semantic relevance rather than merely popularity through cross-linking or advertising.

4. Integration of Social Networks and Shared Web Annotations

Shared bookmarks were chosen as the type of Web annotation to focus on due to their inherent collaborative nature and the popularity of *collaborative tagging* systems such as del.icio.us [20], which allow users to tag Web objects with terms creating shared bookmarks. This kind of metadata is called *observational metadata*, which is generated by users instead of the creators of content. Collaborative tagging is different from most Semantic Web efforts in one important aspect. Whereas the Semantic Web relies on ontologies that are made up of class hierarchies and relationships, collaborative tagging allows users to tag Web objects with any term that they choose. A study of del.icio.us indicates that with only a few tags a consensus emerges for that Web object, constituting a sort of *social proof* [21].

The collaborative tagging approach and its differences with the use of rigid ontologies has been the subject of debate in the Web community over how to best do classification on the Web. Collaborative tagging is seen as lowering the entry barrier since users do not need to understand complex ontologies, and the tremendous effort needed in ontological engineering is no longer needed. However, ontologies have the advantage of reducing ambiguity and the ability to define a hierarchy of terms. It has been observed that users of collaborative tagging systems sometimes attempt to compensate for the lack of a hierarchy by putting hierarchies in a single tag. For instance, a single user on del.icio.us has been observed to use tags such as *blog*, *blog-marketing*, and *blog-stats*.

Popular collaborative tagging shows that users prefer less rigid Web annotation. However, we believe that shared bookmarks and the Semantic Web are not mutually exclusive. General concepts, such as a rating system, may be represented in a formal ontology that is accepted by users who would reject a hierarchical representation of tags.

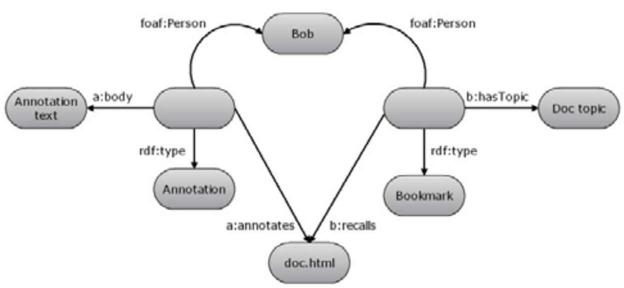

Figure 2. Simplified logical diagram for a shared bookmark connected with a shared annotation. (properties borrowed from FOAF, Annotea annotation namespace, and bookmark namespace.)

One interesting approach for creating shared bookmarks using the Semantic Web is an extension to Annotea [22]. Like the RDF model for Annotea Web annotations, the bookmark model uses classes and properties from other metadata standards in addition to its own standard for bookmarks. The main bookmark properties are *hasTopic* and *recalls*. The *hasTopic* property is used to assign topics to a Web object bookmarked with the *recalls* property. Each Web object can be annotated with more than one topic. Annotea allows the *hasTopic* property to point to topic instances that may have subtopics and may refer to categories in formal ontologies. The notion of topics and subtopics are defined in the Annotea metadata namespace for bookmarks. The RDF model for bookmarks also makes use of properties from Dublin Core to assign titles, identify the bookmark creator, and to associate a longer textual annotation to the bookmark.

Our RDF model for Web annotations, illustrated in Figure 2, is similar to those used by Annotea and Haystack. Like Annotea, the goal is to allow users to use both informal topics and concepts in ontologies to label bookmarks, which can also be attached to a textual annotation. In addition, a simple rating property based on a simple five-point scale is attached to the annotation of Web objects complementing the text annotation. Such a system facilitates searches. For example, a user can search for documents from trusted users using trust values in the social network to find ratings attached to annotations. This type of query would be much more difficult if only a textual annotation was available.

The approach Annotea takes raises a potential issue when users label bookmarks with concepts that do not come from formal ontologies. The fact that users can define topics to be subtopics of other topics means that, in effect, there will be a large number of personalized ontologies alongside with formal ones. While this gives users greater flexibility in organizing their bookmarks, it makes sharing more complex due to peoples' personal hierarchies.

Picture a case where a user wants to find Web objects bookmarked with a term x that is not a part of a formal ontology. Suppose that the user has a personal hierarchy of terms, which includes x. If the query ignores personal hierarchies and returns any Web object labeled with the term x, the user will get objects that are unrelated to his or her definition of x. This is due to the fact that any object labeled with a subclass of x is also labeled with x based on the definition of class hierarchies. Based on the context in which each user creates their personal hierarchies, there may be different interpretations of what should be a subclass of x and what should not be. How bad this is depends on the user's expectation of how the querying system works. If the query doesn't ignore personal hierarchies, we return to the same problem that requires semantic mappings.

Our solution is to provide a namespace for all users that do not intend to use a formal ontology to categorize their bookmarks and annotations. Users would be able to use any term to label their bookmarks, but these terms would not be a part of a hierarchy, as is the case with collaborative tagging. While a user searching for objects labeled with the term x would still see results that are unrelated to what he or she was searching for, this would less likely be seen as bad performance of the system since the user would be aware of the lack of hierarchy. However, this solution forces users to choose between two opposite approaches to categorizing their bookmarks: the collaborative tagging approach or the formal ontology approach. More focused communities would be able to define their own ontologies before using an existing one.

In order to better support collaboration, other users should be able to comment on an annotation. Each annotation should have its own URI and therefore be a resource in itself. Therefore, the annotation model can be used to create meta-annotations, which allows other users to comment and rate another user's annotation. As a result, this model allows the creation of an annotation tree, where entries are responses to an annotation attached to a Web object or responses to annotations. An example of an annotation tree with ratings and comments is shown in Figure 3.

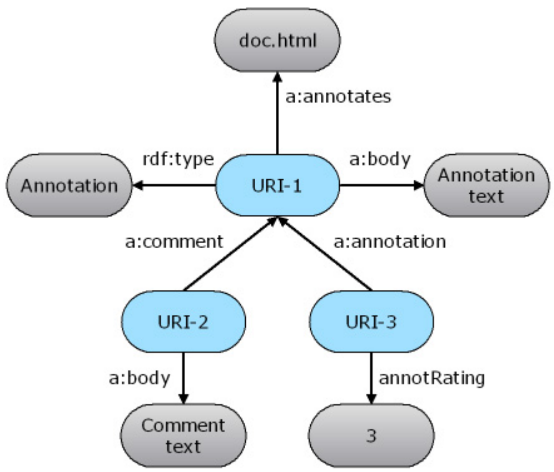

Figure 3. Annotation of a Web object (URI-1) that has two annotations. URI-2 is a user comment for URI-1 annotation. URI-3 assigns a rating for the URI-1 annotation.

Bookmarks are used to categorize and remember a Web object and annotations are used to comment on and rate them. The two are intertwined, and the application should help enforce that. For example, when a user adds a textual annotation or a rating to a Web object, the application should check if a bookmark exists for that Web object and make the connection between the two.

In order to integrate shared annotations with social networks, a foaf:Person object would indicate the owner of a shared bookmark and annotation. Using the extension of FOAF discussed earlier, users would be able to associate a trust value to a read annotation based on the reputation of the creator of the annotation. Metadata from annotations and shared bookmarks have been proposed in a variety of applications [14] such as collaborative spam filtering, user-controlled profiles based on bookmark collections, finding related information using shared bookmarks, organizing search engine results, and a feedback channel for users. In each of these applications, annotations and bookmarks are used for filtering. This combination of annotations with social network information for assessing trust allows users to filter annotations made by unreliable users.

In order for this to be effective, the social network itself must be resilient to bad users generating incorrect trust values. For example, spammers may no longer just flood email inboxes with advertisements but increase their negative impact by tampering social network information. New solutions are needed to create resilient social networks and trust ratings [23].

A key issue in the design of a collaborative system is data storage. Here, RDF documents make up the social network and shared Web annotations and bookmarks. In the World Wide Web, documents can be stored at any server. Therefore, its success can be attributed to search engines that effectively find and index documents. One promising direction for the storage and retrieval of RDF documents is to use a Peer-to-Peer (P2P) network such as RDFPeers [24] which does not use the query flooding mechanism that cause scalability problems. The RDFPeers network is organized in a ring topology and RDF triples are inserted into the network three times, based on applying hash functions to the RDF triple, the subject, attribute, and value, to determine which node the triples are inserted in. For values, a locality preserving hash function is used so that range queries can be performed. RDFPeers supports three types of queries. Atomic triple patterns allow queries on subjects, attributes, and values where each can be a variable or an exact value. Disjunctive and range queries allow queries to limit the domains of variables.

Finally, RDFPeers supports queries that are a conjunction of atomic triple patterns or disjunctive range queries on the same subject variable. RDFPeers guarantees that queries will return results if they exist, does not rely on super-peers, and does not require prior definitions of RDF schemas.

5. Conclusion

In this paper, we extend previous work on shared Semantic Web annotations through exploitation of social network intelligence in the form of shared bookmarks. In order to boost collaboration and community networks through Web-based services, we must ask what kind of meta-information is beneficial in collaborative processes and how it can be effectively generated and mapped across different environments. Furthermore, *trusted* meta-knowledge generation and migration across various applications and platforms will be crucial for collaboration via the Semantic Web. It remains to be seen how legacy and novel collaboration paradigms ultimately integrate with the Semantic Web to boost collaboration effectiveness. We see Web personality ontologies as a building block in leveraging the Web to foster collaboration using richer data semantics. This blueprint for individualized asynchronous collaboration at the intersection with social networking on the Semantic Web is a step in the direction towards richer Web-centric interaction and collaboration.

References

[1] M. Gruninger and J. Lee, "Ontology applications and design," *CACM*, 45(2), 39-41, Feb. 2002.

[2] B. A. Nardi, S. Whittaker, E. Isaacs, M. Creech, J. Johnson, and J. Hainsworth, "Integrating communication and information through ContactMap," *CACM*, 45(4), 89-95, Apr. 2002.

[3] H. Kautz, B. Selman, and M. Shah, "Referral Web: combining social networks and collaborative filtering," *CACM*, 40(3), 63-65, Mar. 1997.

[4] J. Golbeck and J. Hendler, "Reputation network analysis for email filtering," in *Proc. 1st Conference on Email and Anti-Spam*, Mountain View, CA, Jul. 2004.

[5] *The Friend of a Friend (FOAF) project*, www.foaf-project.org/.

[6] P. Mika, "Social networks and the Semantic Web: the next challenge," *IEEE Intelligent Systems*, 20(1), 82-84, Jan. 2005.

[7] G. Antoniou and F. van Harmelen, *A Semantic Web Primer*. Cambridge, MA: MIT Press, 2004.

[8] M. Hatala and G. Richards, "Global vs. community metadata standards: empowering users for knowledge exchange," in *Proc. 1st Int. Semantic Web Conference*, Sardinia, Italy, Jun. 2002.

[9] A. Doan, J. Madhavan, P. Domingos, and A. Halevy, "Learning to map between ontologies on the Semantic Web," in *Proc. 11th Int. World Wide Web Conference*, Honolulu, HI, May 2002.

[10] L. Ding, T. Finin, A. Joshi, Y. Peng, R. Pan, and P. Reddivari, "Search on the Semantic Web," *IEEE Computer*, 38(10), 62-69, Oct. 2005.

[11] J. J. Carroll, D. Reynolds, I. Dickinson, A. Seaborne, C. Dollin, and K. Wilkinson, "Jena: implementing the Semantic Web recommendations," in *Proc. 13th Int. World Wide Web Conference*, New York, May 2004.

[12] J. Kahan, M.-R. Koivunen, E. Prud'Hommeaux, and R. R. Swick, "Annotea: an open RDF infrastructure for shared Web annotations," in *Proc. 10th International Conference on the World Wide Web*, Hong Kong, May 2001.

[13] *Dublin Core Metadata Initiative*, purl.org/dc.

[14] M.-R. Koivunen, "Annotea and Semantic Web Supported Collaboration," in *Proc. 2nd European Semantic Web Conference, UserSWeb workshop*, Heraklion, Greece, Jun. 2005.

[15] D. Huynh, D. Karger, and D. Quan, "Haystack: a platform for creating, organizing and visualizing information using RDF," in *Proc. Int. Workshop on the Semantic Web*, Hawaii, May 2002.

[16] B. Katz, J. Lin, and D. Quan, "Natural language annotations for the Semantic Web," in *Proc. Int. Conference on Ontologies, Databases and Applications of Semantics*, Irvine, CA, Oct. 2002.

[17] D. R. Karger and D. Quan, "What would it mean to blog on the Semantic Web?," in *Proc. 3rd Int. Semantic Web Conference*, Hiroshima, Japan, Nov. 2004.

[18] J. Golbeck, B. Parsia, and J. Hendler, "Trust networks on the Semantic Web," in *Proc. Cooperative Information Agents 2003*, Helsinki, Finland, Aug. 2003.

[19] L. Ding, L. Zhou, T. Finin, and A. Joshi, "How the Semantic Web is being used: an analysis of FOAF documents," in *Proc. 38th Hawaii Int. Conference on System Sciences*, Waikoloa, HI, Jan. 2005.

[20] *Del.icio.us*, http://del.icio.us/.

[21] S. A. Golder and B. A. Huberman, "The structure of collaborative tagging systems," *Journal of Information Science vol. 32, no. 2. April 2006.*

[22] M.-R. Koivunen, R. Swick, and E. Prud'hommeaux, "Annotea shared bookmarks," in *Proc. 2nd Int. Conference on Knowledge Capture Workshop on Knowledge Markup and Semantic Annotation*, Sanibel, FL, Oct. 2003.

[23] J. Golbeck and J. Hendler, "Inferring reputation on the Semantic Web," in *Proc. 13th Int. World Wide Web Conference*, New York, NY, May 2004.

[24] M. Cai and M. Frank, "RDFPeers: a scalable distributed RDF repository based on a structured peer-to-peer network," in *Proc. 13th Int. World Wide Web Conference*, New York, NY, May 2004.