

Using Semantics to Enhance the Blogging Experience

Uldis Bojars¹, John G. Breslin¹, and Knud Möller¹

Digital Enterprise Research Institute, National University of Ireland, Galway
{uldis.bojars, john.breslin, knud.moeller}@deri.org

Abstract. Blogging, as a subset of the web as a whole, can benefit greatly from the addition of semantic metadata. The result — which we will call *Semantic Blogging* — provides improved capabilities with respect to search, connectivity and browsing compared to current blogging technology. Moreover, Semantic Blogging will allow new ways of convenient data exchange between the actors within the blogosphere — blog authors and blog users alike. This paper identifies *structural* and *content-related* metadata as the kinds of semantic metadata which are relevant in the domain of blogging. We present in detail the nature of these two kinds of metadata, and discuss ways of creating such metadata in a convenient and non-obstrusive way for the user, how to publish such metadata on the web, and how to best make use of such metadata from the point of view of a blog consumer.

1 Introduction

Blogs (or weblogs) [18] are online journals or diaries, created by people to express personal or professional views on their world or on observed items that may be of interest to others. Blogs are updated habitually by their creators and are presented in reverse chronological order. There are several popular blogging software publishing tools available at present including Movable Type, WordPress, Blogger and LiveJournal. Movable Type and WordPress are publishing systems which install on web servers to enable individuals or organisations to manage and update blogs and other frequently-updated website content. Blogger and LiveJournal are sites that offer free blogging areas to those who signup to become members.

However, these blogging tools lack the means to add semantics to the blog posts, apart from fixed category topics or free-text keyword tags. Therefore blogs, and the posts that they contain, lack sufficient semantic information regarding the topics that they are talking about or how the current topic under discussion relates to previous blog discussion threads. By augmenting blog posts with machine interpretable metadata, novel ways of both querying and navigating blog information become possible. Metadata about a blog or blog post can be classified as belonging to one of two domains, which we call i) *structure* and ii) *content*. Augmenting a blog with structural and content metadata, as well as the new possibilities which arise from that, is called *Semantic Blogging* [5].

Structure generally speaking refers to the form of a blog. *Structural metadata* identifies and describes things such as the individual parts of a blog (i.e. posts, comments, ...) and their relations, as well as relations between blogs or posts from separate blogs (or any other kind of structured publishing platform).

Complementing structural metadata is *content metadata*, which describes the topic of a blog post — what the post is *about* — i.e. a person, an event, a publication or a webpage. The specific form of content metadata depends on the nature of the topic described. If the topic is a person the blog post talks about, then the metadata for this topic would be that person's name, contact details, etc. If it is an upcoming meeting, then the metadata would be the start and end time, location, etc.

1.1 Example Scenario

Consider the following blogging scenario, as it would probably take place using current web and blogging technology:

Person *A* works in a research group at a university. One day *A*'s supervisor Prof. Gyro Gearloose tells him that he will give a presentation later that week. *A* writes the following post in his blog: *"On friday, 15h there will be a presentation on martian numismatics by Gyro in Room 205. He also wrote an interesting paper on the subject recently, in case you want to read up on it."* He adds links to an official page announcing the event and to Prof. Gearloose's paper.

B comes across *A*'s post on the web. She thinks it's interesting and mentions it in her own blog: *"I read about this really interesting presentation on martian numismatics! I wonder how this relates to cytherian heraldics, so I will definitely go."*

C, being an avid reader of *B*'s blog, decides he also wants to go to this presentation. However, *B* didn't mention any details about the event, and even neglected to provide a link back to *A*'s original post. To get the information he wants, *C* now has to search for "martian numismatics" on the web. With some luck, he finds *A*'s post. He manually enters the details for the presentation into his calendaring application, so that he will be alerted ahead of the event. Also, because he is currently writing his PhD thesis on martian numismatics, *C* downloads the paper and adds the bibliographic details to a database of papers he keeps on his computer. After some searching on the web, he finds out that Gyro is actually Prof. Gyro Gearloose, finds his homepage and enters the professors contact details into his electronic addressbook, in case he has some more questions on the topic.

While, in this scenario, all three participants eventually get where they want, it is a long road and requires a lot of searching and manual copying of data, especially for *C*. In the ideal world, where the Web has become the Semantic Web and Blogging has become Semantic Blogging, the scenario would look slightly different: *A* would attach formal semantic metadata about the presentation (where, when, what, who, ...), Prof. Gearloose (e.g. his contact details) and his paper (bibliographic details) to his blog post. When *B* writes about *A*'s post, she would add structural metadata, indicating that this is a reply to some other post, as well as where to find this post. Finally, *C* would easily get from *B*'s post to *A*'s post, by navigating the follow-up structure which is made accessible through structural metadata. From there, he would simply import the semantic metadata into his own desktop applications, without having to search for them and copy them manually. Spinning the story further, *C* might even have been able to get the metadata directly from *B*'s post (because it is a reply to the original post). Furthermore, another person, let's called her *D*, who is a fan of Prof. Gearloose, would have been able to find *A*'s post directly, by performing a conceptual search for upcoming presentations by Prof. Gearloose, making use of the semantic metadata that is now available within the blogosphere.

1.2 Problems and Solution Formulation

As illustrated by the previous example, the current blogging experience suffers from the fact that there is little or no semantic metadata available in blogs. The topic of a blog post is never made explicit in a machine-interpretable way (with the exception of flat category of tagging systems).

The RSS family and similar newsfeed technologies are the most popular method for syndicating blog posts or obtaining metadata about a blog's internal structure. Syndication allows the copying of one blog's content into another blog (or into a news reader or aggregator). However, these technologies are limited to basic concepts

such as title, description and date as well as a fixed number of the most recently-published blog posts.

In the Semantic Web, meaning can be derived from blogs in a number of ways if advanced blog features are modelled in an ontology and blog data instances are then provided using this ontology. By utilising the existing SIOC¹ (Semantically-Interlinked Online Communities) ontology [2], which was designed to describe a variety of online discussion methods including blogs, we can export more metadata from a server about the internal structure of blog posts. By deriving as much metadata from the underlying blog data stores as possible, connections between concepts can be maintained in a machine-interpretable format for future re-use. Posts on a blog can be linked to their comments, by defining reply connections in either or both directions. The posts can also be linked to the user account that created them, and links between related posts within the blog can be made.

Interlinking within the blogosphere is mainly composed of untyped links between posts and users, as well as trackbacks (a manually created link from one blog post to another, external blog post) and blog rolls (lists of other blogs which a blog author wants to point out). However, trackbacks and blog rolls are limited in that there is no information available on why such a link was created: if the posts are on a related topic; if the original poster is a friend of the referencing poster; if the referencing user agrees or disagrees with the original post; etc.

By using SIOC to materialize the internal structure and also connections between blogs in the global blogosphere, we can harness data across blogs and blogging platforms in new ways. Similar blogs may be linked together, either through explicit links or implicitly through the posts and comments that they contain and the users that created them. Posts may have related posts on other sites, and using SIOC, bidirectional trackback links can be created from the original post to the follow-up. A set of posts can also be formed if they share the same topic resource, or if the topic resources are mapped to each other. Posts and comments by the same user or group of users can be tracked across different blogs. Threaded discussions can be merged or split across blog sites by identifying remote child or parent posts. Most importantly, SIOC is not limited to blog discussions: blog posts or comments can also be related to similar forum threads, Usenet newsgroup postings or mailing list messages if they have been made available using the SIOC ontology.

1.3 Paper Overview

This paper will detail how we can overcome some of the limitations outlined in section 1 and how the blogging experience can be augmented using Semantic Web technologies. We will detail in section 2 how both a blog's content and structure can be described using a number of ontologies. Section 3 will describe the creation steps for this content and structure metadata from a client's desktop and blogging platform's web server respectively. The publishing methods for the content-related and structural metadata is detailed in Section 4, and section 5 will then show how the metadata can be utilised and consumed by Semantic Web applications such as an RDF browser or enhanced blog reader. Section 6 will be a review of related work in this area, and finally in Section 8 we will outline how our work can be further enhanced through custom semantic browsing and querying applications.

2 Metadata

Metadata in the blogosphere formally describes a blog and its individual posts. These descriptions are essentially typed assertions about relations between the blog,

¹ <http://rdfs.org/sioc/>

its posts, authors, other web-resources — or just about anything that can be specified using some unique identifier (specifically, a Uniform Resource Identifier (URI)). The two following sections will in turn look at *structural* and *content-related* metadata.

As regards the choice of metadata format, we suggest the use of Resource Description Format (RDF) as the model for making blog metadata explicit. RDF's graph model makes it much better suited to represent complex objects and relations than the simpler tree structure of XML. Furthermore, since RDF does not impose any specific schema on a given graph and uses URIs as its sole identification scheme, it allows us to integrate data from various sources and conforming to various ontologies or vocabularies. This is especially important with respect to content-related metadata, which can originate from arbitrary sources and be expressed using arbitrary vocabularies. Also, both structural and content-related metadata may well come from different sources and eventually need to be integrated, in order to form a complete graph of blog metadata.

2.1 Structure

As mentioned in the solution formulation, we have chosen SIOC as the ontology for making instances of blog and post structure available. This ontology is described using RDF Schema (RDFS), and instance data is made available in RDF.

Blog Concepts in SIOC The classes in the SIOC ontology of relevance to blogs are *Site*, *Forum*, *Post*, *User* and *Usergroup*, and the main properties linking these classes are shown in Fig. 1.

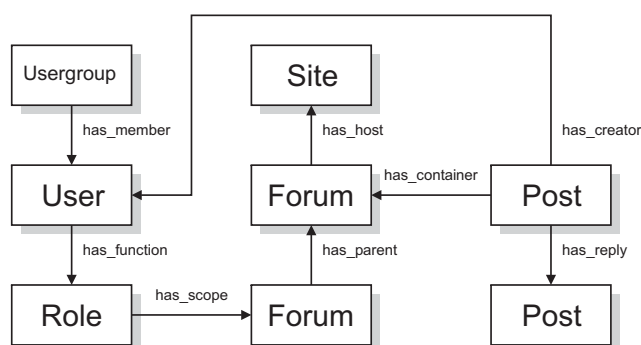


Fig. 1: Main Terms in the SIOC Ontology

Site is the location of an online community or set of communities, and in the context of blogs, it will house one or many blogs. This concept is useful since we can assign a user as the administrator of a site, having moderator control over all blogs hosted at that site.

Forum can be thought of as a channel or discussion area on which posts are made. In the context of blogs, it is a single blog channel. A forum is linked to the site that hosts it. Blog owners can moderate other user's replies to their own blog posts. Blogs may also have a set of subscribed users who are notified when new posts are made.

Post is an article or message posted by a user to a blog. A series of posts may be threaded if they share a common subject and are connected by reply (within a site) or *trackback* (between sites) relationships.

User is an online account belonging to a person who is a member of a community site, such as a blogging area. They are connected to blog posts that they create or edit, to blogs that they can post to or have subscribed to, to blog sites that they administer, and to other users that they know. Users can be organised in a **Usergroup** to control post access to blog areas.

Making Post Connections One of the main use cases for SIOC import involves connecting related post entries between blogs and community sites. Adding SIOC data to posts would open up the connection possibilities as depicted in Fig. 2. Some of these will now be described.

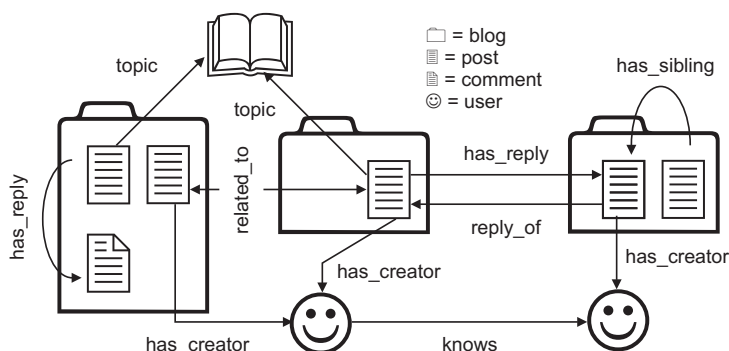


Fig. 2: Structural Relations in the Blogosphere

One of the limitations of trackbacks is that the link is only created in one direction, i.e. from an original post to a referencing post. Using the **related_to** property of SIOC, two posts can be related to each other (and others) in both directions. Apart from an explicit linking of posts, there are other methods of linking blog posts using SIOC, for example, if they share a common topic, creator or container.

A recent development in online discussion methods is an article or post that appears in multiple blogs, or has been copied from one forum to another relevant forum. In SIOC, we can treat these copies of posts as siblings of each other if we think of the posts as non-identical twins that share most characteristics but differ in some manner. For example, a post is created on one blog and categorised with the topic “TV”, but has been copied to another blog with multiple topics such as “Sci-Fi” or “Art”. We can avoid duplication of common data in the creation of siblings by linking to the new sibling, the instance of which only contains the changed properties (in the example, the properties **has_container** and **topic** would change). A number of blog engines support blogging in multiple languages. This leads to multi-language blogs, where the same post can have translations in two or more languages. Related posts across multiple blogs and community sites may also be in different languages. The **has_sibling** property in SIOC can be used for linking these multiple versions or related posts together, with a **locale** property illustrating what language the respective sibling posts are in.

The SIOC ontology allows us to annotate blog posts with topics metadata, allowing the matching of documents on specific topics with each other. While it may be more difficult to require a user to assign a topic to a post at creation time, it is more likely that a forum will have an associated topic or set of topics that can be propagated to the posts it contains. In order to define a topic or category

hierarchy, we propose to use the SKOS framework [1] and create mappings between these concepts and a common category system.

2.2 Content

Content-related metadata describes anything a blog-author wishes to converse about — people, events, books, music, etc. In other words, content metadata covers a very broad domain, especially when compared to the rather specific domain of structural metadata. The exact nature of the metadata will therefore vary significantly between posts: metadata about people might contain their names, homepage or contact-details, a paper might be described in terms of its publisher, title, etc., and an event will have properties such as a start and end time, an organizer, etc. Thus, while it is feasible to define a specific set of concepts and properties to express the whole domain of blog *structure*, it is difficult and problematic to define an ontology to cover all possible blog *content*. What is more, even if one did succeed in defining one all-embracing ontology, it would be very unwieldy and difficult to convince people to use it. Therefore, we propose the use of small, vertical ontologies or vocabularies to describe blog content. Each of these ontologies only covers a certain kind of content, such as people, publications or events. Ideally, one will use ontologies which are already established and widely used — searching, finding and interlinking blog content will then be much easier. In the following paragraphs we will present a number of such small ontologies. All of them are open, well tested and widely used in their respective domain.

FOAF and vCard The Friend of a Friend (FOAF) Project [3] is developing and maintaining an RDFS ontology to describe people, mainly from the point of view of an addressbook context — a person’s name, address, phone number, homepage, etc. The name of the ontology stems from the fact that FOAF also has a means to express whom a person knows, who their friends are. This is achieved by relating one `foaf:Person` instance to another via the `foaf:knows` property. In this way a huge, decentralized network of people — or friends of friends of friends — is established. In a lot of aspects, FOAF is very close to the vCard [7] vocabulary, which covers a similar domain. While vCard doesn’t have the networking capabilities of FOAF, it allows for more detail with respect to specifying addresses. Both ontologies are often used together. vCard is not usually expressed in RDF, but a W3C note exists for representing vCard Objects in RDF/XML².

BibTeX BibTeX [16] is a format for expressing bibliographic metadata, mainly for scientific publications. It is very well integrated in, but otherwise independent from the L^AT_EX system for typesetting. Publications are classified according to types such as *Proceedings*, *Book* or *Article*, and further specified using attributes such as *author*, *title* or *year*, depending on the type of the publication. Like vCard, BibTeX has its own non-RDF representation format, but several implementations in RDF-based ontologies like Semantic Web for Research Communities (SWRC)³ or OntoWeb⁴ exist.

iCalendar iCalendar [8] is an open format for the specification and exchange of event metadata, or, more specifically, calendaring and scheduling data. The iCalendar format has recently gained some attention in the public through Apple’s *iCal*

² <http://www.w3.org/TR/vcard-rdf>

³ <http://ontoware.org/projects/swrc/>

⁴ <http://ontoweb.aifb.uni-karlsruhe.de/>

calendar application — however, even though iCal is built on top of the iCalendar format, the two are completely independent of each other. In fact, iCalendar is supported by a broad range of calendaring applications by other vendors. Similar to the vCard and BibTeX formats, iCalendar precedes the definition of RDF and therefore has its own representation format. However, a W3C workspace⁵ exists that is committed to providing an RDF implementation of iCalendar, as well as a number of tools to provide automatic conversion.

3 Creation Stage

In the previous section we have described the kind of metadata that would be beneficial to find, search for and interlink information from blogs and other resources on the web. In this section, we are going to discuss how such metadata are created. A general requirement for any system that involves the use of metadata is, that the generation of metadata must involve as little work for the user as possible. If Semantic Blogging meant that a blog author had to manually type some RDF/XML code each time they wanted to blog about anything, it would never be adopted beyond a small group of technologically minded people. Even the use of forms to enter metadata would still be far too labour-intensive for Semantic Blogging to achieve any significant impact. Instead, as discussed in Jim Hendler's fundamental article [10], metadata should be generated automatically or semi-automatically while the user performs ordinary tasks they would perform anyways, or even completely without the involvement of the user.

Due to the different nature of structural and content metadata, different strategies have to be applied, as will be discussed in the rest of this section. The strategies for both kinds of metadata are rooted in previous work done by the authors of this paper.

3.1 Structure

Blogs are usually small scale systems consisting of one or more contributors and a community of readers, but their power lies in the large amount of blog data that is available for harvesting. Most blog engines already have RSS export functionality. Since the majority of these blog engines are based on open source software, it is straightforward to modify existing export functions to generate SIOC metadata conforming to the SIOC ontology. In order to retrieve full structural metadata we need to use more of the information available to the blog engine. We have created a plugin for the WordPress blog engine that uses the functions provided by this engine to access its database and export a full set of structural metadata using the SIOC ontology.

The WordPress SIOC plugin⁶ exports information about the main blog data entities — including data about the weblog itself, users creating content, posts and comments that these users have created, topics of these posts and other internal and external structural metadata.

SIOC metadata for blog posts consists of a `sioc:Post` resource and its properties. A URI used to identify the blog post is generated by the blog engine (and is identical to its physical URL). In Table 1 we provide details of the various mappings between entities in WordPress and SIOC metadata, while the central properties used to express the structure of a blog post are described in Table 2.

⁵ <http://www.w3.org/2002/12/cal/>

⁶ <http://rdfs.org/sioc/wordpress/>

Table 1: Mappings of WordPress Concepts to SIOC

Weblog info ->	<code>sioc:Site</code> , <code>sioc:Forum</code> , <code>sioc:Usergroup</code>
Author ->	<code>sioc:User</code>
Posts ->	<code>sioc:Post</code>
Comments ->	<code>sioc:Post</code> linked to by <code>sioc:reply</code>

Table 2: Properties of `sioc:Post`

<code>sioc:has_creator</code>	links a post to a user that created it
<code>sioc:title</code>	contains title of the post
<code>sioc:created_at</code>	creation date and time
<code>sioc:content</code>	contents of the post
<code>sioc:topic</code>	indicates topics or content of the post
<code>sioc:has_reply</code>	links a post to its replies and comments

Additionally, the SIOC properties `related_to` and `links_to` are used to make the connections between posts explicit. They are either extracted from the content of blog posts or inferred from their metadata.

Finally, the WordPress SIOC plugin also creates `rdfs:seeAlso` points consumers of RDF data additional machine-interpretable metadata (e.g. created by a tool like semiBlog).

3.2 Content

As discussed in Sec. 2.2, metadata which describes the content of a post can span over a wide range of domains: metadata about people, events, publications, music, etc. Our previous work on the *semiBlog* blog authoring environment [14] has shown how reusing existing desktop data can be a successful strategy for generating content-related semantic metadata for such a variety of domains. One of the central assumptions is that bloggers will often already have metadata about the topics of their blog available on their desktop⁷. A blog author who blogs about a person will probably already have an entry for that person in their electronic addressbook, someone who blogs about an upcoming event will have this event in their calendaring application, a researcher blogging about an interesting paper might have a BibTeX-entry of this paper available. While a blog author composes a new blog post, the data that already exists in some form on their desktop will be automatically transformed into an RDF graph using an appropriate ontology or vocabulary. E.g., an addressbook entry would be transformed into FOAF, while an event from a calendaring application would be transformed into iCalendar-RDF. The resulting metadata is then attached, turning an ordinary blog post into a semantic blog post.

Reusing metadata from a blog author's desktop in an easy and unobtrusive way requires convenient access to this data. This can best be ensured if the blog authoring environment is implemented as a desktop application. The authoring environment can then e.g. make use of public APIs of the various applications that provide data (electronic addressbook, calendaring application, bibliographic database, etc.), access the system-wide pasteboard for easy drag-and-drop of complex data from these applications or make use of the index of a metadata-enabled

⁷ We use the term *desktop* as a metaphor for the entire working environment within a user's computer.

file system. In the case of semiBlog, application developers can develop plugins for various data sources. Each plugin can accept data from a specific source and knows how to transform it from its proprietary, application-specific source format into a set of RDF triples. This is illustrated in Fig. 3.

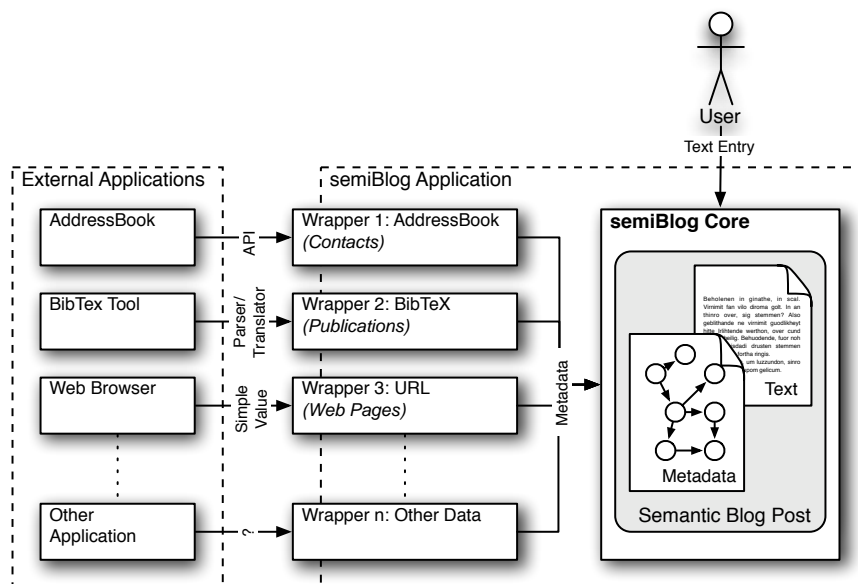


Fig. 3: Creating content metadata from desktop applications

A user can write his blog in semiBlog as they would in any other, non-semantic authoring environment: the application allows to create new posts, write text and format and add pictures. The blog author can then annotate his entry by dragging objects from other applications onto the post in semiBlog. The relevant plugins convert the incoming data and produce equivalent RDF graphs. Finally, these will be merged into one bigger graph, which contains the content-related metadata for the complete post.

Naming the Metadata An interesting general problem arises in the creation stage of content-related metadata: when generating metadata from desktop objects, semiBlog is essentially creating RDF resources which represent the topics of a blog post. E.g., when the post discusses a person A , semiBlog will create a resource which represents A . Each such resource can be assigned a URI to make it uniquely identifiable. URIs ensure that data from various sources can be integrated, since resources with the same URI will be considered identical. The problem, however, is that semiBlog cannot know which URI to choose for each resource it creates — it is not obvious what the URI of any given thing in the world is. We have identified three general naming strategies:

- *Random URI* - Generating a random URI by using a Universally Unique Identifier (UUID) generator algorithm, e.g. [13]. This is the easiest solution and also the one currently employed by semiBlog. However, it breaks the idea that resources which represent the same real world object have the same URI, since different semiBlog instances would create different URIs. Explicit `owl:sameAs` statements could still identify equality, however, this equality would first have to be

inferred in some way. A solution might be rules or heuristics such as the inverse functional property `mbox` used in FOAF, which determines that instances of `foaf:Person` which have the same values for `foaf:mbox` are considered equal.

- *Desktop URI* - Internally semiBlog uses URIs to identify objects on the desktop, from which the semantic metadata will be generated. However, these URIs identify information items, and not the real-world entities which are described by these, and are therefore not used externally.
- *URI authority* - Instead of deciding itself, semiBlog could gather as much information as possible about the resource in question, and forward it to some external service. This service can then determine the URI on the basis of the given information. However, since such a service doesn't exist at the moment, this approach is also not an option.

4 Publishing Stage

Publishing is the stage where semantic metadata is made available to the world, e.g. to a human reader who is accessing the blog through a browser, or an automatic agent which is looking for RDF on the web. The following sections will first briefly describe how content-related metadata generated by semiBlog is prepared and sent to an arbitrary blogging platform. Then we will illustrate how the WordPress SIOC plugin can be used to automatically integrate this data with the structural metadata it produced on a WordPress⁸ installation.

4.1 Preparing Content Metadata

The general strategy for publishing content-related semantic metadata produced with the semiBlog application is very simple: a link is created for each object with which the blog author annotated his post. The link points to some location on the web where the RDF metadata about this object can be found. All links are then added to the bottom of the HTML code of the post and typed as mime-type `application/rdf+xml`, so that they can be recognized and picked up by specialized crawlers (e.g. by the WordPress SIOC plugin, see Sec. 4.2). One major advantage of our approach is that a blog user is not restricted in his choice of a blogging engine, since it is irrelevant whether or not it accepts upload of RDF or any method for publishing metadata at all. In fact, the blogging platform won't even know it just received a semantic blog post — as far as it is concerned, it only received some HTML code. What is necessary, however, is that semiBlog has access to *some* service that can accept and publish RDF (such as an RDF repository like YARS [9]), which it would then link to. An example of a post with added metadata within the WordPress blogging platform can be seen in the screenshot in Fig. 4.

Different blogging platforms require different methods of programmatic access for inserting new posts or change existing ones. Some wide-spread methods are XML Remote Procedure Call (XML-RPC)⁹ based APIs like the MetaWeblog API¹⁰ and the MovableType API¹¹, or Blogger's Atom API¹². semiBlog acknowledges this situation by offering plugin interfaces for publishing blog content (three different interfaces for text, metadata and media content). Application developers can implement these interfaces for any access method they would like to use.

⁸ <http://wordpress.org/>

⁹ <http://www.xmlrpc.com/>

¹⁰ <http://www.xmlrpc.com/metaWeblogApi>

¹¹ <http://www.sixapart.com/movabletype>

¹² <http://code.blogger.com/archives/atom-docs.html>

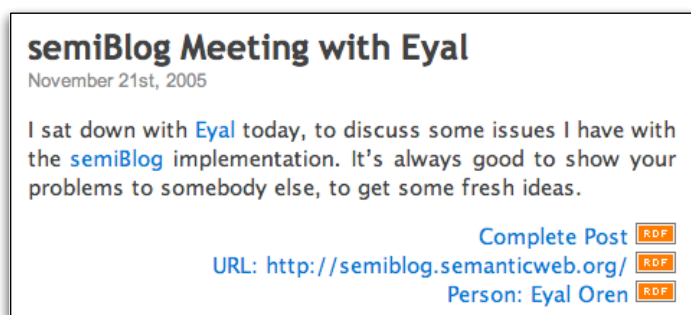


Fig. 4: Screenshot of a blog post with attached metadata

4.2 Integration with Structural Metadata

In the following paragraphs we will show step by step how a blog post enriched with content-related metadata is transferred to a WordPress installation, where it is automatically integrated with the structural metadata produced by the WordPress SIOC plugin.

1. After producing the metadata, semiBlog uploads it to an external service which can receive and publish RDF. This service will provide semiBlog with a URL where this specific piece of metadata is available. Depending on the nature of this service, these URLs could point to files which contain RDF code, or could be complex queries which will return a view on an RDF repository such as YARS.
2. semiBlog adds links to these URLs to the blog post and transfers it to the WordPress installation, using the MetaWeblogAPI through XML-RPC.
3. When activated (by requesting data from a specific PHP script), the SIOC plugin in the WordPress engine will derive SIOC metadata about the desired post from the blog engine (its database and internal logic).
4. As a part of this process, the plugin will extract the URLs of the content metadata from the post's body and link to them using `rdfs:seeAlso` statements. Using `rdfs:seeAlso` is common practice to indicate additional information to a given resource, and allows consumers of the SIOC data to include the metadata provided by semiBlog.
5. The result is a combined graph of structural and content metadata, as illustrated in Fig. 5.

Thus the SIOC plugin integrates all metadata about a blog post and acts as a bridge between the data generated by semiBlog and users of RDF data that would not examine a blog post's body looking for additional *content metadata*. This allows semiBlog to be generic and work with any blog engine, while at the same time it provides an extended functionality if the blog is hosted on a blog engine that has a SIOC export capability.

Independently from producing the actual metadata, the plugin also provides an auto-discovery link on the blog post's HTML page. Such links are common practice to indicate and point to machine-readable metadata for a given web page.

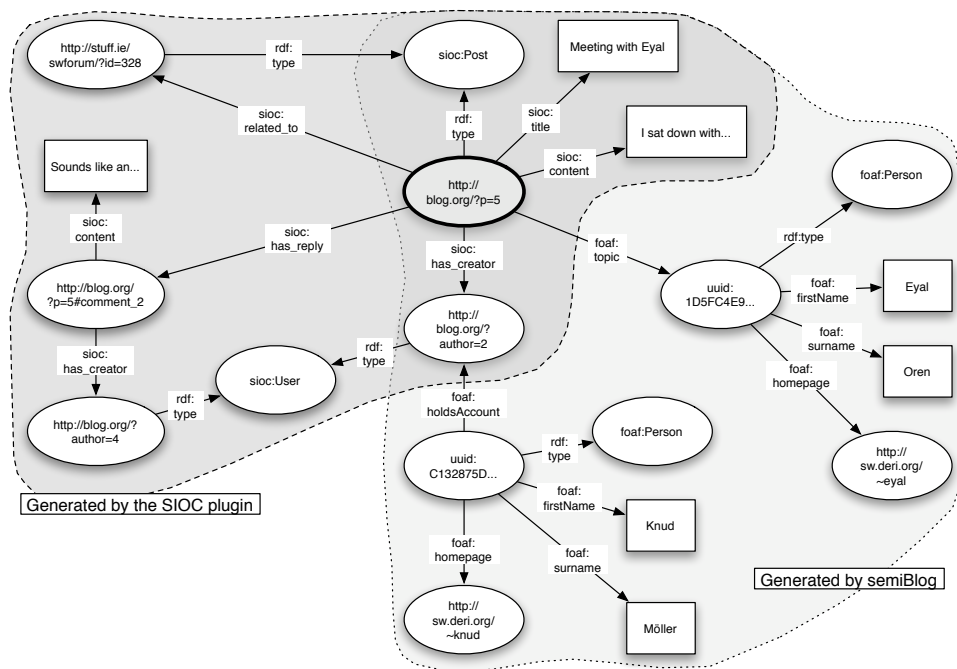


Fig. 5: A combined SIOC and semiBlog graph

5 Consuming Stage

Once semantic metadata has been attached to a blog and made available, there are a number of ways to consume and make use of this data. We will outline some scenarios in this section.

5.1 The Semantic Web as a Clipboard

We have suggested that content-related metadata can be generated by exporting existing desktop data and transforming it into RDF. This can be made to work both ways, as it has been shown in [15]. Using a metadata-aware blog reader, a user can detect metadata attached to a blog post, and import it into his own desktop applications in the way the author had exported it before. This kind of data exchange through a blog could also be described as using the web as a clipboard, and would be very useful as a communication channel within any kind of organizational context.

5.2 Crawling and Browsing the Metadata

Having annotated blog entries with semantic metadata enables the collection, querying and browsing of this information and the blog entries it describes. First we collect metadata, and mirror it in an RDF store, where it will be indexed and probably be enriched with data from other sources. This data can then be used by semantic applications built on top the RDF store. Cf. [2] for a closer look at this scenario.

Crawling The SIOC plugin provides an auto-discovery link, which functions as a starting point for RDF crawlers and applications by indicating where to find RDF data about the blog or a particular post:

```
<link rel="meta" type="application/rdf+xml" title="SIOC"
      href="http://blog.org/wp-sioc.php" />
```

The metadata is then collected by an RDF crawler that recursively traverses `rdfs:seeAlso` and similar links and submits the data into the RDF store. The following steps illustrate in more detail how the crawler works in our case:

1. Use auto-discovery hints to find the URL where the SIOC plugin has published RDF data.
2. Collect RDF data provided directly by the SIOC plugin.
3. Recursively traverse `rdfs:seeAlso` links to crawl RDF data provided by semi-Blog or other sources.
4. Submits data into an RDF data store (e.g. YARS).

Query and Browsing The metadata in the YARS store can then be queried directly using a RDF query language such as N3QL¹³ or SPARQL¹⁴, and be displayed in browser applications that are capable of rendering the raw metadata in a form better suited for human users. We have created the prototype of a node browser, which displays the content and structural metadata stored in the RDF store, showing the links between blog posts and the things they are describing and allowing to navigate these connections by exploiting the RDF graph model. The browser is still under development, but available for testing online¹⁵.

6 Related Work

So-called folksonomies or community-based tagging systems such as Technorati¹⁶ or del.icio.us¹⁷ provide a simple yet effective way of adding content-related metadata to blogs (and web pages in general). However, this flat and string-based metadata clearly lacks the expressive power of an RDF based solution.

A number of recent papers have specifically investigated the topic of Semantic Blogging from different angles. [11] discuss a semantic blogging prototype built on top of the Semantic Web browser Haystack [17]. They interpret blog entries mainly as annotations of other blog entries and web resources in general, and devise a platform to realise this in terms of the Semantic Wev. The paper also underlines the inherent semantic structure of blogs and their entries as such, and presents a way of formalizing these semantics. [4] puts a strong emphasis on the use of semantic technologies to enhance the possibilities of blog consumption, by allowing viewing, navigation and querying with respect to semantics. The paper describes a prototype for both creation and browsing of semantic blogs, which was developed as part of the SWAD-E project¹⁸. While the prototype only deals with bibliographic metadata as annotations to blog entries, the authors point out that the same technologies can be used for any kind of metadata. citeohmukai2004semblog describes a platform called Semblog, which uses the FOAF ontology as an integral part. FOAF descriptions of blog authors are linked to their blogs. In this way, the blog as a whole is annotated with metadata about its author. On a more fine-grained level individual blog entries are classified by linking them to personalised ontology. To implement their platform, the authors provide both a Perl CGI-based tool called RNA and a standalone Windows-based tool called Glucose.

¹³ <http://www.w3.org/DesignIssues/N3QL.html>

¹⁴ <http://www.w3.org/TR/rdf-sparql-query/>

¹⁵ <http://rdfs.org/sioc/browser>

¹⁶ <http://www.technorati.com/>

¹⁷ <http://del.icio.us>

¹⁸ <http://www.w3.org/2001/sw/Europe/>

7 Future Work

When creating a new post, it would be useful if as well as being able to copy and paste content from a blog post that the desired content annotations could also be transferred. Also, as well as information about the content, post references could also be dragged and dropped into a new or edited post (for example, to create a trackback or related to link between posts) to create typed links between posts. This is along the lines of the RDF clipboard idea by Tim Berners-Lee¹⁹.

Leveraging the full potential of SIOC requires the provision of custom programs and user interfaces specially tailored towards browsing SIOC data. In the consuming stage, we discussed how the Node Browser application can be used to navigate and search for aggregated information from both a blog's content and structure - similarly other RDF browsers such as BrownSauce²⁰ could be used. However, it would be useful to have a more graphical method for browsing not only this information, but also to allow one to navigate from a post to its related posts or "distributed conversations" across different blog sites. This could be a "SIOC explorer" application that would allow users to browse SIOC-enabled sites transparently without need for data warehousing, simply by traversing `rdfs:seeAlso` and other links in RDF. A similar open source application already exists: Foafscape²¹ is a browser for navigating FOAF-related RDF data that uses the Prefuse²² visualisation toolkit to display hyperlinked graphs of Friend of a Friend data, and this could easily be modified for navigating SIOC data.

Another aspect of future work is in relation to the argumentative nature of blog discussions, in a similar way that [graphical] issue-based information systems ([g]IBIS) [6], [12] examined the argumentative nature of design and planning discussions. At first glance, a user is unable to tell if a blog post and resulting discussion is overall supporting or opposing the topic(s) being discussed. For example, a person is researching medicine X for which they have a prescription, but they only want blog discussions on the negative aspects of X (already knowing the advantages). It would be desirable to provide details of an argumentative structure so that the associated meaning of the proposition/counter-proposition synthesis could be instantly recognised when browsing blog discussion topics. Some reply types such as agree or disagree have been ontologised by the W3C²³, but these may be augmented with some level or scale of agreement.

8 Conclusions

This paper detailed a means for enhancing a user's blogging experience by leveraging the fusion of two kinds of metadata related to blog posts: content-related and structural. We described ways of creating such metadata in a convenient and non-obtrusive way for the user, by dragging and dropping object annotations from a user's desktop and by instantiating structural metadata that is automatically created during the blogging process. We detailed how such metadata can be published on the web through a popular blogging platform, and finally we described how metadata can be reused in posts or utilised for cross-site browsing by a blog consumer.

¹⁹ http://www.w3.org/2001/sw/Europe/reports/xml.sw.prototype_math_logic/#Part1

²⁰ <http://brownsauce.sourceforge.net/>

²¹ <http://foafscape.berlios.de/>

²² <http://prefuse.sourceforge.net/>

²³ <http://www.w3.org/2001/12/replyType>

9 Acknowledgements

This material is based upon works supported by the Science Foundation Ireland under Grant No. SFI/02/CE1/I131.

References

1. A.J.Miles, N.Rogers, and D.Beckett. SKOS Core RDF Vocabulary, 2004. <http://www.w3.org/2004/02/skos/core/>.
2. J. G. Breslin, A. Harth, U. Bojars, and S. Decker. Towards Semantically-Interlinked Online Communities. In *The 2nd European Semantic Web Conference (ESWC '05), Heraklion, Greece, Proceedings, LNCS 3532*, pages 500–514, May 2005.
3. D. Brickley and L. Miller. FOAF Vocabulary Specification. <http://xmlns.com/foaf/0.1>.
4. S. Cayzer. Semantic Blogging and Decentralized Knowledge Management. *Communications of the ACM*, 47(12):47–52, December 2004.
5. S. Cayzer. Semantic Blogging: Spreading the Semantic Web Meme. In *XML Europe 2004, Amsterdam, Netherlands, Proceedings*, April 2004.
6. J. Conklin and M. Begeman. gIBIS - A Hypertext Tool for Exploratory Policy Discussion. In *The Conference on Computer-Supported Cooperative Work, Proceedings*, pages 140–152, 1988.
7. F. Dawson and T. Howes. vCard MIME Directory Profile, 1998. RFC 2426: <http://www.ietf.org/rfc/rfc2426.txt>.
8. F. Dawson and D. Stenerson. Internet Calendaring and Scheduling Core Object Specification (iCalendar), 1998. RFC 2445: <http://www.ietf.org/rfc/rfc2445.txt>.
9. A. Harth and S. Decker. Optimized Index Structures for Querying RDF from the Web. In *3rd Latin American Web Congress, Buenos Aires, Argentina, Proceedings*, pages 71–80, October 31 to November 2 2005.
10. J. Hendler. Agents and the Semantic Web. *IEEE Intelligent Systems*, 16(2):30–37, March/April 2001.
11. D. R. Karger and D. Quan. What Would It Mean to Blog on the Semantic Web? In S. A. McIlraith, D. Plexousakis, and F. van Harmelen, editors, *Third International Semantic Web Conference (ISWC2004), Hiroshima, Japan, Proceedings*, pages 214–228. Springer, November 2004.
12. W. Kunz and H. W. J. Rittel. Issues as Elements of Information Systems. Technical Report WP-131, University of California, Berkeley, 1970.
13. P. Leach, M. Mealling, and R. Salz. A Universally Unique Identifier (UUID) URN Namespace, 2005. RFC 4122: <http://www.ietf.org/rfc/rfc4122.txt>.
14. K. Möller, J. G. Breslin, and S. Decker. semiBlog - Semantic Publishing of Desktop Data. In *14th Conference on Information Systems Development (ISD2005), Proceedings*, Karlstad, Sweden, August 2005.
15. K. Möller and S. Decker. Harvesting Desktop Data for Semantic Blogging. In *1st Workshop on the Semantic Desktop at ISWC2005, Galway, Ireland, Proceedings*, pages 79–91, November 2005.
16. O. Patashnik. BibTeXIng, February 8 1988. BibTeX Documentation.
17. D. Quan, D. Huynh, and D. R. Karger. Haystack: a Platform for Authoring End User Semantic Web Applications. In *Second International Semantic Web Conference (ISWC2003), Proceedings*, 2003.
18. J. Walker. Weblog. In D. Herman, M. Jahn, and M.-L. Ryan, editors, *Routledge Encyclopedia of Narrative Theory*, page 45. Routledge, London and New York, 2005.