Towards a New Generation of Social Networks:
Merging Social Web with Semantic Web

Liana Razmerita
(Copenhagen Business School, Denmark
lr.isv@cbs.dk)

Rokas Firantas, Martynas Jusevičius
(IT University of Copenhagen, Denmark
rokas@itu.dk, martynas@itu.dk)

Abstract: This article investigates several well-known social network applications such as Facebook, LinkedIn, Last.fm, Flickr and identifies semantics and social data portability as the main technical issues that need to be addressed in the future. We argue that these issues can be addressed by building social networks like Semantic Web applications with FOAF, SIOC, and Linked Data technologies, and show how this can be done by implementing a prototype application using Java and core Semantic Web technologies. The developed prototype shows how features from semantic websites such as Freebase and DBpedia can be reused in social applications and lead to more relevant content and stronger social connections. In addition, the article discusses a number of issues that need to be addressed in the design and implementation of semantic social networks, which could serve as a guideline in the implementation of other semantic social applications.

Keywords: social networks, Semantic Web, Linked Data, semantic interoperability, ontology
Categories: H.1.2, H.5.1, H.5.2, H.3.4, H.5.3, H.5.4

1 Introduction

Social web or Web 2.0 has fundamentally changed the way in which people interact, communicate and share information. The on-line collaboration, interaction and participation is facilitated by social networking sites (LinkedIn, Facebook) and social applications (YouTube, Flickr) where the social dimension, and in particular the user community tightly integrated into the application domain is crucial for the success of the application. User communities are created around various domains like music sharing, photos sharing, videos sharing, social bookmarking sharing, professional networks, etc. On the Social Web, users are provided with means to express their views and are encouraged to provide content. User-generated content and metadata is provided in simple forms like tagging, ratings, comments, blogging, podcasting or wikis. Social networking applications are driven exclusively by the content provided by the users, which is one of the defining characteristics of the Social Web. As a result large amounts of data and information are gathered in “walled websites”. Relevant information is often poorly structured, buried in low-quality information or surrounded by highly subjective data. Most of the social networking applications are “walled websites” and the “online communities are like islands in a sea” [Bojars, ‘08].

Despite their tremendous success, social networking websites have a number of limitations that are identified and discussed below. Lack of interoperability between
data and lack of semantics in different social network applications limits access to relevant content available on various social networking sites, and limits the integration and reuse of available data and information. This may result in a growing dissatisfaction of the user community and a reduced usability of the websites. This article shows through the implementation of a prototype that Semantic Web technologies can be used to build a next generation of social networks that overcome limitations of current social network applications and enable new features currently not exploited by them. The article emphasizes the need for semantics and mechanisms to better structure this information and make it interoperable.

Research combining social networks and the Semantic Web is an interdisciplinary field, attracting researchers from both social and computer sciences. More research combining social networks and the Semantic Web is required to address the above-mentioned limitations. An important line of research combining social networks and the Semantic Web focuses on the extraction of semantic data from existing social applications, its representation and its analysis. Existing work in this area explores the possibilities of extracting ontologies from user contributed folksonomies through collaborative tagging systems and of integrating ontologies with folksonomies [Specia, '07; Xu, '06], while other approaches propose the development and evolution of lightweight ontologies in a collaborative way [Angeletou, '07; Mika, '07]. Researchers seem to agree that folksonomies and lightweight ontologies have more properties in common than differences and will be further integrated, and thus, in the future, a community-based bottom-up approach might prevail over top-down controlled engineering efforts. Much of the current research for representing simple user profiles is based on the Friend of a Friend (FOAF) project\(^1\) – a project aimed at “creating a Web of machine-readable pages describing people, the links between them and the things they create and do”. FOAF is currently an important source of RDF data available on the Web which has already been used for social network analysis [Ding, '05; Finin, '05; Paolillo, '05]. A related initiative is the Semantically-Interlinked Online Communities (SIOC) project,\(^2\) which provides an ontology for describing items and relationships from Internet discussion methods (such as blogs, forums, and mailing lists) to facilitate interconnection of these methods by publishing metadata [J. Breslin, '07; J. G. Breslin, '05]. Many recent papers highlight a growing interest in portability issues among social network applications; they are regarded as “fundamental problems”, and semantic technologies, mainly FOAF, are being proposed to solve them [Mika, '05].

Theoretical work combines the Semantic Web (SW) and social networks, especially for the analysis of social networks and the extraction of knowledge from existing data [Ding, '04]. However, neither the creation of new end-user semantic social applications nor their design and implementation is well explored. Existing social network applications do not employ SW technologies, although most of the standards infrastructure is already in place. Most are “walled” websites, which provide limited means for users and developers to control, publish, and access social data. This limits possibilities for reuse and integration, which are the driving forces

---

1 [http://www.foaf-project.org/](http://www.foaf-project.org/)
2 [http://www.sioc-project.org/](http://www.sioc-project.org/)
behind both Web 2.0 and the Semantic Web, and results in growing dissatisfaction in the user community.

This article discusses some of the limitations of current social network applications and shows how Semantic Web technologies can be used to build a new generation of social networks which overcome these limitations and enable new features currently not exploited. The article is structured into six sections. The section below discusses a number of common features and technological limitations of current social networking applications. Section 3 identifies and describes a number of issues that need to be handled in the design and implementation of the next generation of semantic social networks. Section 4 presents a concrete scenario usage for a semantic social application. A presentation of a semantic social network prototype featuring semantic mashups is proposed in section 5. The last section summarizes the role of semantics for the development of a new generation of social networks in order to better harness and integrate the collective knowledge made available by the various social applications.

2 Study of Social Networks

Social networking sites are developing at a very fast pace, attracting more and more users. These sites have become central points on the Web for sharing personal information and online socialization. They allow users to create a personal profile and link to profiles of their friends. The resulting network can be browsed to find common friends, friends that have been lost or forge potential new friendships based on shared interests [Mika, 07].

The core feature in all social networks is the user community, which is tightly integrated with the application domain. We have analysed applications which we personally use and which we think reflect the current state of the art in social networking: Last.fm, Flickr, YouTube, Facebook, LinkedIn. As an outcome of a review of social networks, we have identified a number of generic features that are common to the majority of social networks as well as a number of technological limitations.

Common features of social networks include:
- User-contributed content - The content is provided exclusively by the users.
- Recommendations - If a user shows interest in some resource, the application suggests other resources that might be relevant or similar.
- Folksonomies - Users are allowed to tag almost any resource.
- Comments - Users are allowed and encouraged to comment on almost any resource.
- Top lists can be of various types: lists of most interesting, most popular, most commented on, most recently added or updated resources and so on. Lists can be compiled on a personal or group basis, or for the whole website.
- Favourites are either built by users manually by adding resources, or compiled from the resources the user has shown interest in.
- User groups - Users can join groups, whether public or restricted. They are usually set up by people with similar interests or a circle of friends.
- Data accessed using Application Programming Interfaces (API). As a recent Web 2.0 trend, social networks have begun to open up some of their data to
the public, which is later most often used in various mashups. Most sites publish RSS or custom XML data feeds, while others provide programmatic methods to control the site to the same extent as using the graphical interface.

Social networking applications have a number of technological limitations as summarized below:
- It is not possible to export/import profile data from one application to another
- It is not possible to export/import social relationships from one application to another
- There is usually less data available in machine-readable formats than the application contains
- Application Programming Interfaces (APIs) are based on a variety of custom formats and protocols, some of which are non-standard (such as Facebook Query Language (FQL)—a language similar to SQL and Facebook Markup Language (FBML) which is HTML with custom elements in Facebook).

Our observations fit well with statements by initiatives such as Open Social Web, Social Network Portability, DataPortability, OpenID, OpenSocial, which have emerged as a result of a growing dissatisfaction in user communities.

Semantic social networks will still focus on the community dimension while drawing on Semantic Web technologies to aggregate content. Some examples of semantic social networks are further discussed in section 2.1.

2.1 Examples of Semantic Applications

**Freebase** is an open database of the world’s information. It acquires structured data spanning different domains such as music, people and locations from various sources such as Wikipedia and MusicBrainz. The data is aggregated, identical or related concepts being linked together. In addition, users in the community can add, edit, and even upload data. Topics in Freebase are organized by types which are grouped into domains. An important feature is that users can not only fill already predefined types with instance data or edit it, but can also create their own types and define their properties, i.e. they can create new schemas and extend Freebase’s domain model using the same interface. Furthermore, it provides an open but proprietary API for its data and encourages its use in applications and mashups.

**DBpedia** is a community effort to extract structured information from Wikipedia and to make this information available on the Web via a semantic representation. It provides an RDF dataset extracted from Wikipedia, which contains mostly free text but also structured information such as categories, lists, infoboxes, links to external resources.

---

3 http://opensocialweb.org/
4 http://microformats.org/wiki/social-network-portability
5 http://www.dataportability.org/
6 http://openid.net/
7 http://code.google.com/apis/opensocial/
8 http://musicbrainz.org/
pages etc. DBPedia makes it possible to perform complex queries (such as “German musicians who were born in Berlin”) over a SPARQL query interface. DBPedia is a prime example of Linked Data publishing and can be browsed using semantic browsers. It is interlinked with other semantic datasets such as Geonames, MusicBrainz etc.

3 Towards Semantic Social Network Applications

The next generation of Social network applications using semantic technologies needs to address a number of issues that are identified and further discussed, e.g. the creation of metadata, user interface design, openness of domain models, the publishing and reusing of data and metadata. These four aspects are further elaborated below.

3.1 Creation of metadata

Folksonomies use uncontrolled keywords (tags) like a simple form of metadata to annotate and categorize items on a website, making them easier to discover and retrieve. Folksonomies are the primary sources of metadata on Web 2.0. However, they have issues with consistency, ambiguity and lack of synonymy. A next step beyond Web 2.0 is the Semantic Web. Semantic Web or Web 3.0 is a new layer of the Web, a “Giant Global Graph” [Berners-Lee, 2006], which will make use of semantics and access machine processable data in order to better support navigation, searching and provide a number of enhanced services for the users. On the Semantic Web, creation of metadata, a data with machine-processable meaning, is as important as the data itself. Metadata derived from tag clouds and folksonomies may help aggregating various fragmented pieces of data and information into collective knowledge. It has been observed how folksonomy tags evolve into property:value triple-tags, which serve the same purpose as subject property object triple statements in RDF and thus folksonomies move towards becoming lightweight ontologies. This phenomenon has even been called “poor man’s RDF” [Andersen, '05]. Social networks will provide more sophisticated means to directly create RDF metadata, and collaborative tagging may evolve into lightweight ontology development and may be integrated into collaborative modelling of the social network domain.

3.2 User interface

Much work is left on the issue of how to present semantic data to the user in applications, not to mention editing it [Ankolekar, '08]. There are a number of semantic browsers, such as Tabulator, Disco, OpenLink RDF Browser, Objectviewer, Zitgist. They are able to render generic RDF data for human users

---

9 http://www.geonames.org/ontology/
10 http://www.w3.org/2005/ajar/tab
11 http://www4.wiwiss.fu-berlin.de/bizer/ng4j/disco/
12 http://demo.openlinksw.com/rdfbrowser/index.html
13 http://objectviewer.semwebcentral.org/
14 http://dataviewer.zitgist.com/
and navigate through different data sources through RDF links, just as conventional Web browsers navigate through HTML links. However, this kind of presentation is probably too advanced for mainstream Web users (see Figure 1).

**Figure 1: Tabulator view**

It can be assumed that a Semantic Web application interface visualizes its domain ontology in such a way that each class and instance has its own page, linked to others through class-instance and instance-instance relationships. This *generic* approach is used in many semantic websites, and is probably best illustrated by Freebase. Another approach, which we call *specific*, is used by conventional Web applications, as well as social networks. Every type of information (such as a car, a user, or an event) has its own specific user interface. For each new type a new interface has to be created; the same interface cannot be used for different types, and interfaces have to be fixed when the schema changes. This approach is obviously not feasible on the Semantic Web, where ontologies are meant be extended, reused, and integrated from different sources. If social networks are to become extensible semantic applications, it is likely that they will have to adopt the generic approach.

### 3.3 Domain model

Social network applications (Last.fm, Flickr, LinkedIn, etc.) are usually developed for different application domains such as music, photos, and business. However, they share a common property: the domains are fixed and non-extensible. Users are encouraged to contribute and improve application data, but this is restricted to instance data for predefined types. Semantic applications such as Freebase take a different approach and allow users to edit the domain model itself: not only to fill in instance data, but to extend and edit types, add new types, and define properties in the underlying ontology. Following this approach, social network applications would empower users to express their identities by creating or reusing concepts and relationships relevant to them, and share them with others. The domain model could be left to the community to control and further develop it in a direction which is currently of most interest to it, keeping it relevant over time. People would connect
through things they have in common, achieving object-centered sociality [Breslin, 07]. In the future, this may be achieved by integrating lightweight ontology development into the means of user collaboration and content contribution. In order to implement this approach, applications are usually modelled in RDF/OWL as they offer more expressivity than object-oriented and relational models. This data model is based on formal semantics and therefore interpreted unambiguously by different agents. Furthermore, they need to reuse FOAF and SIOC ontologies, which are currently the state of the art representations of social networks on the Semantic Web, as well as other relevant ontologies. Most current SW applications are also static and fixed in the sense that ontologies are known and mapped manually at design time [Razmerita, 03]. Although semantic technologies are designed with extensibility and openness in mind, current programming languages and tools are not able to fully exploit it. It is expected that future semantic applications will be using multiple ontologies, discover them and integrate them on request.

3.4 Publishing and reusing data and metadata

Large amounts of meaningfully interlinked RDF data available on the Web are crucial for achieving the Semantic Web vision. However, many social networks do not offer interfaces and APIs to access application data. Others make the contents of the website (such as lists of users, songs, or pictures) available via a simple read-only REST interface in a software-processable data format, usually a custom schema of XML, Atom, or RSS. Some provide full APIs with add/update methods, invoked via various interfaces such as REST, XML-RPC, SOAP, Atom, or OpenSocial. A variety of publishing formats (especially non-standard) make reuse difficult. We advocate that semantic social networks should publish their data in RDF, designed specifically for distributed knowledge representation. Furthermore, all resources in social networks (including non-information, “real-world” resources) should be given URIs, distinguished from URIs of the representations that describe them, and published as Linked Data. APIs should be replaced by SPARQL endpoints, which would allow running remotely structured queries against application data. Semantic data representation and advanced interfaces would help to overcome portability issues of proprietary APIs and interconnect social networks with different data sources, enable use of semantic browsers, and facilitate semantic mashups.

4 Use case scenario

In this section, we provide a concrete scenario of how a semantic social network would enhance current Web 2.0 tools and the user experience. Consider Sarah, who is a socially active person both online and offline. She likes to go out several days a week and hang out in bars or clubs. She also likes art and cinema, and visits exhibitions and movie theatres once in a while. Sarah is also picky and busy, so she likes to evaluate all the cultural and night-out options she has, preferably planning them at least a week in advance and only visiting those which she finds interesting. Sarah is an advanced Web user, but she is still struggling to find information about

http://linkeddata.org/
what events are there, and where and when they take place. Most of the information is online, but it is spread across many different websites and is also incomplete. In a club-goers portal she usually finds a list of parties for the next week. However, it is compiled by editors from the official announcements received from clubs, and consequently many smaller venues or underground events are left out. For an overview of current exhibitions and opening hours, she has to visit homepages of several museums, check another portal to see the movie timetable, and finally put all the dates and times on one list to see how they match her schedule. Sarah often likes to try something new, so even when she finds an event of interest, she may have no previous knowledge of the venue it takes place in. Therefore she has to look up the address in yet another website, a mapping service, to find her way there. Now, imagine she comes across a website where users themselves are allowed and encouraged to contribute content about events and places. Places are listed by type (such as Club or Museum) and location, which can be shown on the map. Events are also shown by type (such as Party or Exhibition), and are related to place and time. Sarah can immediately see a list of events of any type, at any place, or any date, as well as a list of places at a certain location. Moreover, users are able to indicate which events they are attending and which places they usually prefer, and on the basis of this lists of the most popular events and places are built, which might help her to make her decision. She might subscribe to a feed of upcoming events and follow it in her RSS reader. If Sarah wants to contribute to the site or make use of the social features, she has to sign up and create her profile with some personal information. After that, she can add friends as well as favourite events and places to her profile. From an analysis of events and places she and her friends like, the application might suggest some new possibilities. If Sarah knows an event will be taking place but is not listed on the website, she can simply fill in a short description form, set the place for it, and save it. Since the website is built with semantic technologies, Sarah is also able to import friend profiles from other social networks that publish them in a semantic format. She is also able to run structured queries against the data in the website and put a list of events she will be attending in a mashup on her homepage.

5 Prototype application design and implementation

The application prototype is a social network that allows users to browse events (such as concerts and conferences) and places of interest (such as venues and hotels) and find those that are most relevant to them. As described in the previous section, the purpose of this application is to make it easier for the users to discover events and places that are most likely to be of interest to them.

The prototype features a generic user interface. Users are able to browse OWL ontology classes and their instances, and see properties with values, as represented in Figure 2. Within the application users are able to import a list of friends from an external FOAF source. The application provides Linked Data access by serving interlinked RDF/XML which can be visualized and browsed using semantic browsers such as Tabulator, or reused in other applications. It provides a SPARQL endpoint, which allows running structured queries. It also implements a semantic mashup: when a page of an instance is requested, the application queries remote DBPedia SPARQL
endpoint, retrieves its description and homepage address in real time and presents it to the user in the same fashion as local properties and values. The prototype is implemented based on a RESTful Web framework which treats HTTP resources as first-class objects and follows a Model View Controller (MVC) pattern and W3C standards.

On top of the data layer, Model-View-Controller (MVC) pattern is often used for logical separation of application domain and presentation layers [Krasner, 88].

Figure 2: Browsing the ontology in a Tabulator view

5.1 Model View Controller (MVC) design pattern

Model View Controller is a design pattern used in software engineering. In complex computer applications that present lots of data to the user, one often wishes to separate data (model) and user interface (view) concerns, so that changes to the user interface do not impact the data handling and the data can be reorganized without changing the user interface. The MVC design pattern solves this problem by decoupling data access and business logic from data presentation and user interaction, by introducing an intermediate component: the controller. The controller translates interactions with the view into actions to be performed by the model. In a stand-alone GUI client, user interactions could be button clicks or menu selections, whereas in a web application they appear as HTTP GET and POST requests. The actions performed by the model include activating business processes or changing the state of the model. Based on the user interactions and the outcome of the model actions, the controller responds by selecting an appropriate view.
Within the prototype, the Model is the ontology layer, the Java code is the Controller and Views are generated by integrating SPARQL queries results and transforming them into XHTML using XSLT. The application domain is modelled as an RDF/OWL ontology stored in a RDF triple store, accessed using Jena\(^{16}\) and queried using SPARQL. The domain ontology extends classes such as foaf:Person and adds a number of new classes such as: Place and Event as represented in Figure 3. FOAF and SIOC classes and properties are reused. Views become representations of REST resources (XHTML, RDF). They join several SPARQL XML results and transform them directly to output XHTML using XSLT, or serve raw RDF/XML for the Linked Data interface, depending on the HTTP *Accept* header. Controller dispatches requests to resources which have explicit URIs, implements HTTP methods, and can be related to domain instances using *foaf:topic*, and return view representations.

Most of the current object-oriented languages are statically typed and do not allow classes to be changed or extended at run-time. Thus it is not easy with the existing tools to map Event class in OWL to an Event class in Java so that it can be changed or extended at run-time.

Tools such as RDFReactor\(^{17}\) and Elmo\(^{18}\) generate object-oriented Java code from our ontologies, but this code is static and not extensible at run time and therefore it was not used.

### 6 Conclusions and future work

This article has provided an overview of the state of the art of actual social network applications. Common features and technological limitations were identified and

---

18 [http://www.openrdf.org/about.jsp](http://www.openrdf.org/about.jsp)
discussed in section 2. Based on this analysis, in section 3, the authors propose four main areas (creation of metadata, user interface design, open domain models, publishing and reusing data and metadata) that require further investigations and developments in order to implement semantic social networks. Semantics is the key for overcoming current limitations of social applications. Furthermore, social data portability issues are leading to dissatisfaction in both user and developer communities. This is caused by limited amounts of social data published openly and lack of tools to import them as well as formats and APIs of limited interoperability. Social networks would benefit from Semantic Web technologies. FOAF, SIOC, and Linked Data can solve portability issues and enable data reuse.

The prototype described in section 5 shows how features from semantic websites such as Freebase and DBpedia can be reused in social applications and lead to more relevant content and stronger social connections.

New generations of social applications may take advantage of the advanced data model that SW technologies provide. Semantic data representations and advanced interfaces would help to overcome portability issues of proprietary APIs, interconnect social networks with different data sources, enable the use of semantic browsers and facilitate semantic mashups.

Domain models could be collaboratively developed by users of the application. This approach requires a new generic user interface based on classes, instances, and properties. This could lead to more up-to-date and relevant content, which in turn would facilitate social connections through points of common interest. Another interesting approach that could be further explored in the future is the use of an AJAX-enabled application interface, a form-based interface for SPARQL and dynamic, run-time object-ontology mapping tools.

Acknowledgment
A short version of this article has been presented at the Adaptation and Personalisation for Web 2.0 conference, associated with User Modeling, Adaptation and Personalisation 2009 conference.

References


