Social Resource Promotion

A semantic approach for managing resources with object-centered social networks in the Web of the future.

Timm Heuss
Darmstadt University of Applied Sciences, Darmstadt, Germany
Timm.Heuss@{stud.h-da.de, web.de}

Wednesday 31st August, 2011

Keywords: Object-centered social networks, Folksonomies, Social Web, Web 2.0, Semantic Web, RDF, XMPP, mDNS, JAX-RS, Smack API, Jena Semantic Web Framework

Abstract

Since the beginning of the Web, URIs were considered the most important basic technology of the Web. Within the realms of the Social Web, communities formed to tag, comment and rate such resources in so called folksonomies. However, nowadays, these communities have developed in technically distinct and inoperable walled gardens, violating the paradigms that were once associated with the Web.

To tackle these issues with the best practices of the Social and the Semantic Web, this paper proposes an approach for the automatic exchange of resource descriptions via object-centered social communities - built on an XMPP and mDNS networking stack. The resulting infrastructure concept called Social Resource Promotion (SRP) will provide an automatic and semantic way for peer interoperability and enable everyone to leverage his very own object-centered, highly customized information mix.

The concept of SRP was successfully positioned at the Federated Social Web Workshop 2011 in Berlin, which also confirmed major parts of SRP's technology stack and architectural principles.

Taking these results as well as recent developments into account, this paper continues the development and the realization of a prototype implementation named SocIO, bringing to live the key ideas of SRP.

The resulting software is publicly available, hosted on the social coding platform GitHub, and constitutes the starting point for further development by the author or the community.

1 Introduction

When Tim Berners-Lee invented the World Wide Web (WWW), he addressed very practical, ordinary issues of his working environment in the European Organization for Nuclear Research (CERN) laboratory these days: Because it was a "technological melting pot" [6, p. 19], sharing and exchanging research results across different operating systems and data formats was a major problem within the organization. Berners-Lee considered the diversity of computer systems and networks as a potential resource - not as a problem - and had the goal to break out of local barriers to create a cross-computer, free and "living world of hypertext" [6, p. 27]. In this spirit, featuring fundamental properties like a powerful referencing system and complete decentralization [6, p. 15f], the Web was born.

2 The beginning of the Web

According to Berners-Lee, the Web is built upon three basic technologies, which form "common rules" (in decreasing order of importance): Uniform Resource Identifiers (URIs)[1], Hypertext Transfer Protocol (HTTP) and Hypertext Markup Language (HTML) [6, p. 36]. While HTTP and HTML allow every user to write and publish information of any kind worldwide, URIs, as the most important basic web technology, provide "a simple

[1]The term URI can be "further classified" to an Uniform Resource Locator (URL): In addition of identifying resources, URLs also *locate* resources in terms of a "primary access mechanism" or a "network location"[5, p. 6].

and extensible means for identifying a resource" [5, p. 3] in a "global scope" [5, p. 5].

It is important to note that these early approaches of the Web consisted only of these three technologies, allowing the Web to be as simple, as decentralized and as interoperable as possible [6, p. 36f].

As the Web became more and more popular, it allowed more and more new use cases. Some of them, including the commercial ones, contributed new technology to the Web's development. An example is the so called e-commerce in the middle of 1990s, which required a way to transfer credit-card purchases securely and thus motivated the development of the Secure Socket Layer (SSL) [6, p. 97].

These developments made the Web to the universal place it is today, however, they relied on the same three basic building blocks described above. And because especially HTML is designed to be read and understood semantically by humans - not by machines - development mainly focussed on the so called "eyeball Web": the Web targeted at humans [10, p. 82].

3 The Semantic Web

To enable machines to benefit from the huge success the Web's has for humans, the World Wide Web Consortium (W3C) introduced the *Semantic Web* in the 2000s. In order to make information available to machines, an extending and complementary set of technologies was introduced [7].

The base technology of the Semantic Web is the data format Resource Description Framework (RDF). Aligned at the so called *AAA slogan* that "Anyone can say Anything about Any topic" [3, p. 35], it defines a structure that is meant to "be a natural way to describe the vast majority of the data processed by machines"[7].

RDF expresses meaning by encoding it in sets of *triples* [7], composed of `subject` (corresponding to an entity or thing, like people, places or other concrete objects), `predicate` (properties of the entities) and `object` (assigned entities or literal values like a string or a number)[19, p. 19].

As figure 1 shows, RDF forms a *taxonomy* by defining a "collection of terms being used in a particular domain, that can be structured (e.g. hierarchically)" [10, p. 58].

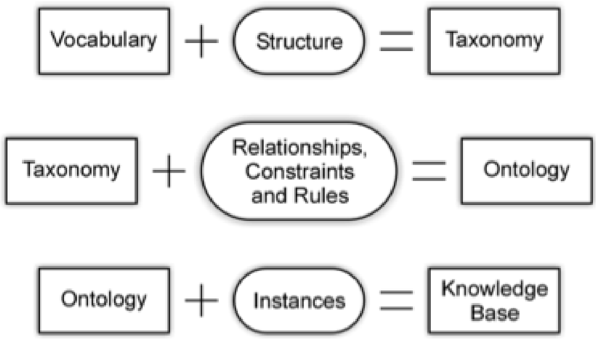

Figure 1: The big picture of the terms taxonomy, ontology and knowledge base as fundamental concepts behind Semantic Web technologies. [10, p. 58].

The subsequent term is an *ontology*, which is a "understandable, useful, durable" semantic model [3, p. 1] or a "document [...] that formally defines the relations among terms" [7]. In the definition of Breslin et al and as depicted by figure 1, an ontology is a taxonomy enriched by relationships, constraints and rules.

A *knowledge base* is ultimately formed by an ontology and concrete instances.

Among RDF, the W3C has defined standards and tools, covering knowledge base, ontologies and taxonomies, which are, however, not in the scope of this paper[2].

Aligned at the ideas of the early Web, the Semantic Web is designed "as decentralized as possible" [7]. Because of the interoperable and open data formats, it enables machines to collaborate and to transfer data among themselves - without an artificial intelligence and without being "expressly designed to work together". [7].

[2] To read more about Semantic Web technologies, see the presentation slides of Prof. Dr. Bernhard Humm to his Semantic Web lecture available at `https://www.fbi.h-da.de/organisation/personen/humm-bernhard/sw0.html` (URL last access 2011-08-21), his origins in the book Semantic Web for the working ontologist by Dean Allemang and Jim Hendler [3] and the big picture views of Breslin et al. in The Social Semantic Web [10].

```
Tagging(object, tag, tagger,
        source, + or -)
```

Figure 2: Thomas Gruber's theory behind the term folksonomy describes a relation of the parameters object, tag, tagger, source and a simple polarity [13].

4 The Social Web / Web 2.0

Despite the efforts of the W3C to promote the Semantic Web, it suffered from the so called "chicken-and-egg problem": "it is difficult to produce data without interesting applications, and vice versa." [10, p. 71]. In addition, Tim O'Reilly describes the issue that the "extra task" in adding semantics to information by "building hidden structure into the data" has "no immediate benefit to the user" [17].

So instead of moving towards a machine readable Web, the Web since the middle 2000s mainly opened to a broader audience, formed the *Social Web* or the *Web 2.0*. By lowering technical barriers, activities like "collaboration and sharing between users" [10, p. 12] became common. Emerging trends like Blogs or Wikis are prime examples for this development. In addition, the popular term Web 2.0 - originally defined by O'Reilly[3] - express "new structure and abstractions [...] on top of the ordinary Web" [10, p. 11].

The role of the user changed from being a "consumer of content" to an "active participant" [10, p. 22]. This new role is best demonstrated by the emergence of a concept called *folksonomy*, which was originally pioneered by services like del.icio.us[4] and flickr[5] [16]. According to the popular definition by Thomas Vander Wal [21], a folksonomy is the "result of personal free tagging of information and objects (anything with a URL)" in a "social environment" - "tagging that works".

More formally, Thomas Gruber defined a folksonomy to be a tagging with five parameters [13] (see figure 2):

`object` is the tagged resource, `tag` the word used to describe the resource and `tagger` expresses "the person or agent doing the tagging" [13] for collaborative environments. Furthermore, because it can't be guaranteed that `object`, `tag`, `tagger` are used consistently across various systems, Gruber added `source` to express the dependency to a concrete system. The concluding parameter `+ or -` allows one to express "polarity" [13] in the relation.

A closer look shows that Gruber's theory of a folksonomy is behind many elements of the Social Web - even if they don't focus on all parameters of this relation. For example, del.icio.us emphasizes the relation between `object` and `tag`, while more recent developments like Facebook's Like-button or Google's +1 seem to concentrate on creating ways to express polarity.

5 Issues of current approaches

Tagging resources (URIs) is a standard feature of the Social Web and still "common to many social websites" [10, p. 38]. But the manner it is done today ambiguous, as described in the following.

Over time, many services have developed their own term describing the concept of providing and maintaining folksonomies - or at least a simplified version of it. Figure 3 shows an example of the webpage O'Reilly Radar, which embeds three common social services to allow readers to promote articles.

In the terms of a folksonomy, those buttons all do the same: They allow users to express and to share their affinity to a resource. Facebook calls this "Like", Twitter "Tweet", Google "+1". In addition, these services give users usually the possibility to add some custom words to resources, like comments or tags.

However, these services are technically very different. This difference leads to the inhomogeneous situation of the current Social Web and is interconnected with a lot of issues concerning original Web paradigms, which are described in the following sections.

5.1 Limited interoperability

Being interoperable - once a fundamental key feature of the Web - and making money with a service is likely to become a clash of interests in the Social Web nowadays. Especially Social Networking Services (SNSs) - services that allow "a user

[3] `http://www.oreilly.de/artikel/web20.html` (URL last access 2011-08-1).

[4] `http://www.delicious.com/` (URL last access 2011-08-01).

[5] `http://www.flickr.com/` (URL last access 2011-08-01).

Figure 3: "Social Web Buttons" in the today's Web. Each service adds his very own button with its own underlying API, its own closed circles of users, its own terms of use. Image source `http://radar.oreilly.com/2011/07/maker-faire-2011.html` (URL last access 2011-08-01).

to create and maintain an online network of close friends or business associates for social and professional reasons" [10, p. 169] - "make money off of [their users] data" and have an interest "to lock in [the user's] data so that [he] can't move between services and leave them in the dust" [8].

It is important to note that the issue is not the commercial interest of SNSs or the business model of individual services like Facebook. The issue is that those commercial interests lead to technical monocultures and limited interoperability instead of extending the possibilities of the Web like, for example, the e-commerce did when it motivated SSL (see also page 2).

Some of the SNS providers even developed competitive, proprietary formats to open standards: the news streams available in Facebook and Twitter are in state of competition with the open Really Simple Syndication (RSS) format [9]. And it is not surprising that exactly those providers actively limit or hide their RSS capacities to promote their own format [9].

That's why those providers are sometimes refereed as "walled gardens" [8] - isolated islands with closed users groups, closed features and content. Even though some of these technologies are connected and able to interoperate, the majority of them "do not usually work together" [10, p. 39] - causing the users to maintain our pool of beloved and trusted data manually, across a huge diversity of tools and on a regular basis.

5.2 Single source of power

Another related issue is the fact that SNSs become a central source of power and can allow or disallow whatever they want. Beyond the purely technical problem of having a single point of failure in those systems, the social life becomes more and more dependant of the terms of use of those providers, which are moreover subject of frequent changes.

And it already happened that those terms restricted technical possibilities. For example, when Facebook blocked the browser extension Facebook Friend Exporter[6], which allows users to "grab all the information about [their] Facebook friends", the company claimed that the extension violates the terms of use [18].

So users are encouraged to maintain and document more and more parts of their private lifes in SNS, but are legally or technically limited in exporting back their own, originated data.

Providers might argue that users have the free choice either to accept the terms of use or not. But is there a real choice, when parts of your social life depend on it? If you can't participate at all if you disagree?

A recent study gives hints on the answer: As Netpop Research revealed[7], 85 percent of the users[8] are at least uncertain about their privacy in Facebook - 47 percent are even concerned. It is a significant indicator how much social pressure surpasses individual privacy concerns, when 8 out of 10 users use a service they don't trust.

Especially by the example of SNS like Facebook as one of the most popular services in the Web, we're far beyond the point Berners-Lee stated in [6, p. 133]: The Social Web is no longer "so huge that there's no way any one company can dominate

[6] `https://chrome.google.com/webstore/detail/ficlccidpkaiepnnboobcmafnnfoomga` (URL last access 2011-07-28).

[7] `http://www.netpopresearch.com/node/26713` (URL last access 2011-07-27).

[8] User base is U.S. broadband users who used Facebook in the last 30 days.

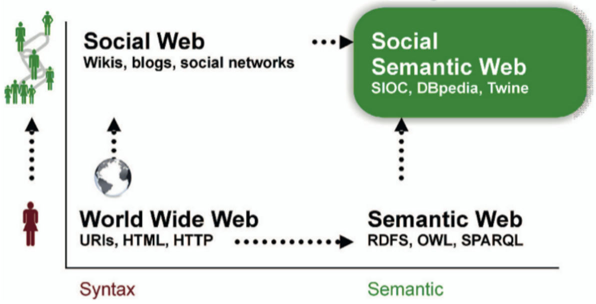

Figure 4: The Social Semantic Web is meant to combine the Social and the Semantic Web [10, p. 15].

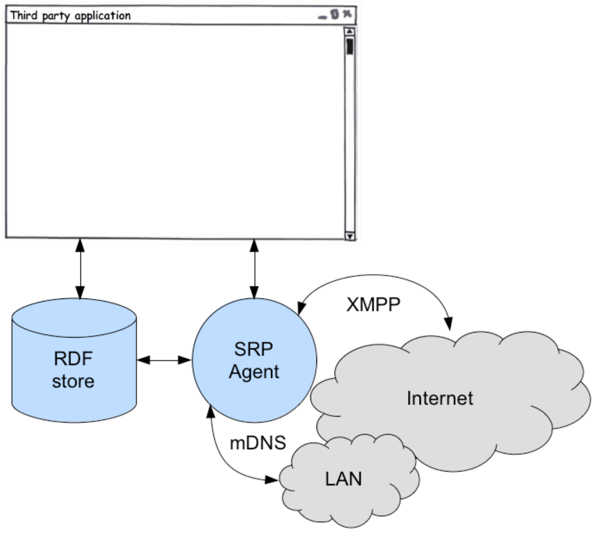

Figure 5: Architecture of a SRP-enabled application, including a RDF store and a SRP agent, both providing APIs for third party applications, and a two-protocol networking stack.

it". In the contrary, the activity of a single company affects millions of users.

Interoperability and decentralization are no more key principles of the Social Web and the Web's fundamental visions of the early days are in danger.

6 The solution: SRP

To tackle the issues described above, the author has developed the concept called Social Resource Promotion (SRP). It is technologically aligned as the fusion of the Semantic and the Social Web. As figure 4 depicts, this combination of "a network of interlinked and semantically-rich knowledge" is sometimes refereed as *Social Semantic Web* [10, p. 14].

SRP is a central infrastructure component which manages the entire ecosystem of URIs. This includes actions like the storage and annotation, as well as commenting and tagging. Furthermore, SRP can be organized in several *object-centered social networks* - networks (or "groups") of people with a common interest [10, p. 39] - where people can connect freely and only driven by their interests.

The key feature of SRP is the exchange of *promotions* - semantic descriptions of folksonomies for certain resources. By utilizing its social connections, SRP gives the option to share content between human peers or computers (in form of users or agents) and within a well defined, selected number of relevant peers. SRP thus facilitates everyone's personal, highly customized URI-pool, wired into several social communities.

Unlike the popular distributed, open-sourced SNS Diaspora[9], SRP focusses on the transport mechanism of semantic folksonomies, while Diaspora targets on building a ready to use, distributed SNS including a frontend for end users. Acting as back end for various third party applications, SRP thus enables software to benefit from a huge, highly customized and specialized information mix.

6.1 Architecture

Figure 5 depicts the architecture which follows the common design principles of a semantic application, described by Dan Allemang and Jim Hendler in [3, p. 59f, 64-74], picking up the idea of the *network stack*, described by John G. Breslin et al. in [10, p. 194-196].

6.1.1 Core components

The core is formed by a RDF store to persist URIs and folksonomies, which are expressed in one or more onotologies, and an application, the *SRP agent*, operating on it.

[9]`https://joindiaspora.com/`
(URL last access 2011-08-21).

While the RDF store would usually have its own (query) API , depending on the usage pattern it might be also useful to provide a *native* API for third party applications, based on the observer pattern[10]. This enables applications to register listeners for RDF-`predicates` and `objects` and thus use SRP, without being forced to implement Semantic Web technologies like RDF and Simple Protocol And RDF Query Language (SPARQL) directly.

6.1.2 Network stack

To connect and to collaborate with peers, the network stack is built by two basic technologies:

The first is *Extensible Messaging and Presence Protocol (XMPP)*[11] (also known as *Jabber*), a robust, commonly supported, widely used and decentralized instant messaging protocol. XMPP's fundamental roster concept[12] allows a very differentiated maintenance of peers on the *buddy list*, thanks to the subscription model. As shown on in figure 6, this concept can be utilized for social networking.

The second basic technolgy of the network stack is Multicast DNS (mDNS)[13], a serverless protocol for local networks with various implementations (such as Apple Bonjour[14] or Avahi[15]). Using mDNS, the first contact negotiation is simplified and automated, taking advantage of the fact that people with common interests usually come together physically.

6.2 Interoperability

It makes perfect sense to use RDF as a description medium for such promotions, not only because it was literally developed to *describe resources*, but because there already exist a number suitable ontologies for tagging [10, p. 144-148] and describing people including their relations [3, p. 169-177].

In addition, it is stated that ontologies and folksonomies are in fact "complementary (and synergistic) paths towards enhancing the Web" [10, p. 143].

Listing 1: Contact initialization message (RDF/-Turtle format) sent by the SRP agent. [19]

```
<http://xmlns.com/foaf/0.1/Agent>
   foaf:jabberID <xmpp://sttiheus@h-da
      .de> ;
   srp:capable "true"^^<http://www.w3.
      org/2001/XMLSchema#boolean> ;
   foaf:interest <http://dbpedia.org/
      page/Computer_science>.
```

In a first, conceptual approach, the Friend of a friend (FOAF)[16] and the Tag[17] ontologies are used. However, thanks to the very nature of RDF, those ontologies might be (ex-)changed as usual to cover other, more complex datasets and usage scenarios.

In the following, the contact initiation and the exchange of promotions are described.

6.2.1 Establishing social networks

Initiating the first contact follows pretty much the FOAF ontology. As listing 1 shows, such a message could contain an identification (in form of a Jabber Identifier (JID), `foaf:jabberID`) to automatically add and subscribe the agent on the XMPP roster, a SRP-specific flag (like `srp:capable "true"`) to indicate capability and a basic set of interests for an early classification:

This first request might be sent manually via XMPP by people who met online or it might be broadcasted to capable SRP clients in the local network using mDNS. To prevent spamming incapable clients, SRP clients should be identified by an appropriate XMPP resource identification[20] respectively an appropriate mDNS presence message[21]. Additionally, further control flags might indicate the desired notification scheme.

[10] `http://en.wikipedia.org/wiki/Observer_pattern` (URL last access 2011-04-30).

[11] `http://xmpp.org/xmpp-protocols/rfcs/` (URL last access 2011-04-27).

[12] `http://xmpp.org/rfcs/rfc6121.html#roster` (URL last access 2011-04-28).

[13] `http://www.multicastdns.org/` (URL last access 2011-04-27).

[14] `http://developer.apple.com/opensource/` (URL last access 2011-04-27).

[15] `http://avahi.org/` (URL last access 2011-04-27).

[16] `http://xmlns.com/foaf/spec/` (URL last access 2011-04-27).

[17] `http://www.holygoat.co.uk/projects/tags/` (URL last access 2011-04-27).

[20] See chapter *Resource binding* in the XMPP Core Specification (RFC 6120), `http://tools.ietf.org/html/rfc6120#section-7` (URL last access 2011-04-27)

[21] XMPP Serverless Messaging (XEP-0174) specification is highly relevant in this matter, `http://xmpp.org/extensions/xep-0174.html` (URL last access 2011-04-30).

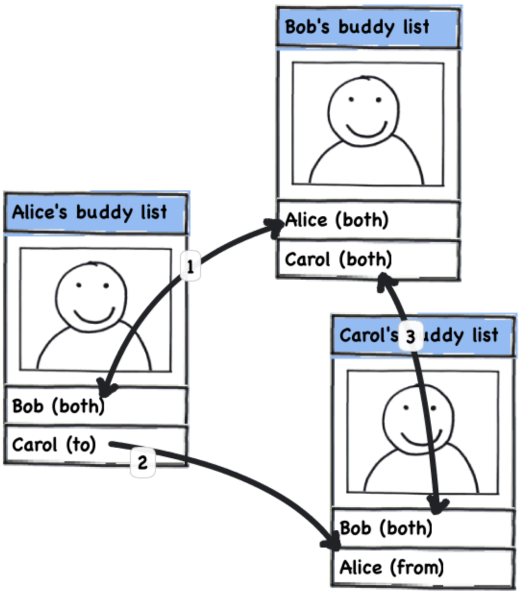

Figure 6: Using the XMPP-subscriptions to fine-tune the object-centered social networks: Alice and Bob have subscriptions to each other (1), Carol and Bob, too (3). Alice ended her subscription to Carol and thus she doesn't receive promotions anymore, while she still sends own promotions to her (2).

After two agents are connected by their XMPP rosters and form a small social network, the fine-grained XMPP-subscriptions[22] can be used to do further customization, as figure 6 shows: While the default setting might be a mutual subscription (like Alice and Bob, Bob and Carol), one might always choose to disable promotions from certain peers (like Alice did for promotions from Carol).

This enables users to build up trusted object-centered networks, purely driven by their interests and their own quality considerations.

6.2.2 Exchanging folksonomies

The actual exchange of folksonomies is, due to the existing Tag ontology, as straight as the contact initialization. Sending the respective RDF statements to all authorized users on the roster could

Listing 2: Folksonomie of the URI `http://fbi.h-da.de/`, including JID, tag assignment, timestamp and a postive user vote. [24]

```
<http://fbi.h-da.de/>
 tags:tag [
   tags:associatedTag tag:
       officalDepartmentPage ,tag:
       reliable ;
   tags:taggedBy [ foaf:jabberID <xmpp
       ://sttiheus@h-da.de> ] ;
    tags:taggedOn "2011-04-27T10
        :43:00.000Z"^^xsd:dateTime
 ] ;
 srp:likedBy [ foaf:jabberID <xmpp://
     sttiheus@h-da.de> ] .
```

be triggered when the user just performed a certain *activity* - like assigning a keyword to a URI, bookmarking it, et cetera. Furthermore, more *content*-focussed actions might be quite useful too, e.g. when the content of a certain URI is (in the user's opinion) right or wrong. In this context, Thomas Gruber has coined the term *Negative Tagging* [13], describing the idea that a tagging should, besides of the tagged object, the tag itself and the source consist of a personal, positive (+) or negative (-) vote.

Listing 2 shows an example how such a promoted folksonomy could look like using mainly the Tag ontology - and once a custom ontology (prefixed with `srp`) in regard of Thomas Gruber's simple rating.

However, one challenge is the design of a comprehensive native API for the SRP agent to reflect the use cases mentioned above. As mentioned, a first approach for that API could implement the well known observer pattern, so third party applications are able to observe certain RDF triples through the SRP agent.

6.3 Application examples

Taking RDF's flexibility and the often referred *network effect* [3, p. 9],[10, p. 114] into account, there are a lot of possible scenarios, in which third party applications benefit from SRP.

[22] See section *Subscription Attribute* in the XMPP Instant Messaging Specification (RFC 6121), `http://xmpp.org/rfcs/rfc6121.html#roster-syntax-items-subscription` (URL last access 2011-04-28).

[24] Prefix definitions of the FOAF (`http://xmlns.com/foaf/spec/`) and the Tag ontology (`http://www.holygoat.co.uk/projects/tags/`), as well as the declaration of the used tags, omitted.

Imagine a SRP-enabled web browser, which advises relevant pages to interested people right after they are bookmarked. Furthermore, changes on webpages could be detected and distributed, as well as moved resources cause an update to be sent to all peers. In the same way, peers could promote warnings for malicious resources.

SRP-enabled news aggregators could provide relevant feeds, or even single posts. Trends and statistics within the connected community could evaluate them and add or remove context relevant or irrelevant resources automatically.

Even the desktop could benefit from SRP: Like a file browser, suggesting community-approved alternatives to existing applications.

6.4 Conceptual conclusion

SRP forms the technical platform for a number of promising developments, as it is meant to reflect the Web's dynamic and speed, while being aligned at the visions of decentralization and interoperability of the early Web. It supports the emergence of URI-pools with high quality information and a maximum of diversity, aligned at the early visions of the Web. With SRP, an incredible number of third party applications can easily socialize and benefit from the knowledge of a specialized crowd.

This enables the automated creation of trusted, decentralized social networks of peers, sharing information of common interests. Thanks to the power of RDF, the kind of information is not limited and can easily be customized, even if a first proposal uses the FOAF and the Tag ontology.

The networking stack consisting of two protocols allow sustainable, decentralized, open-standard communication via Internet with XMPP, and helps users help to benfit from their physical proximity in finding others with common interests.

The described concept of SRP has been submitted and accepted[25] at the Federated Social Web Conference 2011 in Berlin[26], positioned as "[p]olicy-based [approach] to the Social Web that allow one to communicate and share data across specified target audiences" [14, p. 1].

[25] `http://d-cent.org/fsw2011/wp-content/uploads/fsw2011-Social-Resource-Promotion.pdf` (URL last access 2011-08-24).

[26] `http://d-cent.org/fsw2011/` (URL last access 2011-08-08).

7 Discussion

Taking the concept described above as starting point, the following sections will discus and develop the idea by taking a closer look at workshop results and recent developments.

7.1 XMPP as transport protocol

The architectural decision to use XMPP as basic transport protocol is widely confirmed by a number of other projects positioned at the workshop.

Especially because of its federation features and security, the OneSocialWeb project[27] utilizes XMPP as communication protocol for their microblogging framework [12, slide 6]. However, Diana Cheng and Dan Appelquist experienced XMPP's complicated specifications, buggy and bad scaling server software and a need for better Java implementations as drawbacks for the protocol [12, slide 15].

The project of Benjamin Carrillo and Julia Anaya aims at building WebID authentication features in XMPP[28]. They emphasize XMPPs social networking features like contact management and end-to-end connections and claim the protocol to be a "natural choice for implementing a federated infrastructure for social online activities" [11, p. 1]

Iosif Alvertis et al. utilize XMPP to build a "federated architecture to publish messages and retrieve comments under dedicated nodes for specific campaigns"[4, p. 4] and praise XMPP's handiness and real time capacities [4, p. 3]. In addition, because of XMPP's lack of conventional "Social Media APIs that run over HTTP", they think about extending XMPP with so called *Social Connectors* to archive collaboration between XMPP and HTTP/JSON [4, p. 3].

The heavy use of the XMPP protocol confirms the envisioned architecture of SRP.

7.2 Local device detection

With mDNS respectively its implementation in Avahi or Bonjour the author originally intended to detect other SRP agents in the local network environment and to automatically interconnect with

[27] `http://onesocialweb.org/` (URL last access 2011-08-08).

[28] `http://xmppwebid.github.com/xmppwebid/` (URL last access 2011-08-08).

them, as written on page 6. However, at the W3C workshop a project with a quite similar intention was presented:

The Webinos[29] context framework is an EU-founded research project "ensuring that the technologies for describing, negotiating, securing, utilizing device functionalities [...] are fit for purpose" [1]. Intending to "translate the success of the web as a distributed document publishing system into a successful, distributed applications platform" [2, p. 2] the project combines "existing and already used" "state of the art" technology in one solid framework [2, p. 8].

Among W3C and other open standards and specifications, Webinos relies on mDNS for its services to "be advertised so that they can be easily discovered" [2, p. 12] and will be release as open source.

The author expects great benefits from the platform, however, according to Webinos' project schedule[30], a first release will be available in March 2012. Therefore, within the prototype the author will not continue his own mDNS device location approaches until further notice.

7.3 Automated friending

In the concept, SRP agents are able to send a contact initialization message ("Hello"), as listing 1 on page 6 depicts. This shall establish a connection between two peers with a common interest, either on demand or automatically.

The need for an (automated) friending mechanism is also identified by Story et al. On the base of existing protocols to "help automate the weaving of the distributed conversations that make it up" [20, p. 1] - like Track- or (Semantic) Pingbacks - Story et al. introduced *RESTful Pings*, which support semantics, are extensible and work with HTTP return codes [20, p. 4].

RESTful Pings could be also used to wire together SRP agents, in place or in addition to the contact initialization mechanism mentioned before. While the protocol the concept mentioned primarily, mDNS, is only capable for local area networks, RESTful Pings could be a suitable solution for Internet communications.

[29] `http://webinos.org/` (URL last access 2011-07-27).
[30] `http://webinos.org/faq/` (URL last access 2011-08-10).

7.4 Semantic technologies

In order to be as flexible and as open-minded as possible, the concept was designed to work with semantic technology, especially RDF. Therein, the original approach made use of the FOAF and the Tag ontologies (see section Interoperability 6.2 on page 6) to describe and exchange resources.

A reviewer of the position paper suggested the use of the NiceTag[31] ontology by Limpens et al. [15]. NiceTag reconsiders a tag to be "a link between a tagged resource and a sign used to tag" - to cover, as a "key feature of the social Web", the many purposes it serves today [15, p. 1].

NiceTag is powerful in expressing the "multiplicity of facets" [15, p. 7] of tags. The key feature is, however, the fact that with NiceTag, a certain source can be assigned to triples describing a tagging, thus it transforms usual RDF triples to quads [15, p. 3]. While SPARQL is able to query such quads with the keyword `GRAPH`, native RDF does not support " a mechanism to specify the source of each triple" [15, p. 6]. This leads to constrains in the deployment base as all approaches are limited to XML-based notations, so is the W3C Member Submission "RDF/XML Source Declaration"[32]. Unfortunately, this also limits the application of RDF frameworks.

NiceTag is a promising ontology, closely aligned at the author's vision of a general use folksonomy exchange infrastructure. Because of its capabilities to express relationships in a more fine grained way, it seems to fit perfectly into the SRP concept. However, at least for the prototype it is questionable if the benefits of the use of named graphs respectively RDF quads justify the limitations connected to their usage. Therefore, NiceTag will not be employed in the prototype.

7.5 Position to Google+

About one month after the W3C program committee published the author's position paper[33],

[31] `http://ns.inria.fr/nicetag/2010/09/09/voc.html` (URL last access 2011-08-10).
[32] `http://www.w3.org/Submission/rdfsource/` (URL last access 2011-08-22).
[33] `http://d-cent.org/fsw2011/agenda/papers/` (URL last access 2011-08-27).

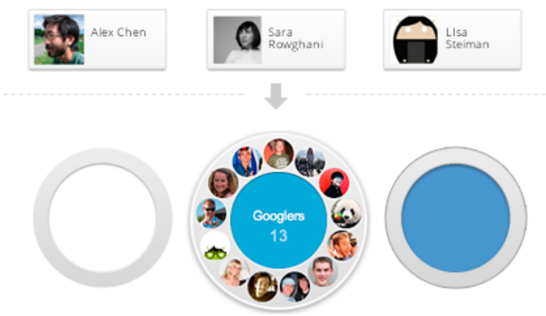

Figure 7: Just as written in the concept of SRP [14, p. 4], Google Circles allow users to connect asymmetrically [22, p. 77]: Every user can assign his contacts to personalized groups ("circles"), independent from the assignee's attitudes. Image source: `https://www.google.com/+/learnmore/` (URL last access 2011-07-27).

Google launched[34] their interpretation of a social network, Google+[35]. While Google's network obviously competes with Facebook and thus both competitors share features, some other features are new to social networks.

7.5.1 Asymmetric connections

A both popular and important feature of Google+ is *Circles*[36]. With Circles, users can assign other users to groups, represented as circles in the management interface [22, p. 77]. Those groups are defined by every user by his own and are highly personalized. Once a user is assigned to a circle, he is notified and can choose what to do with his new contact - whether to add him to one of his circles or to ignore it. These *asymmetric connections* fundamentally differ from Facebook's approach of *Becoming a friend*, where both parties have to agree to a friendship and are subsequently mutually befriended [22, p. 77].

Letting the users build asymmetrical connections was also an original idea of the author's SRP position paper: In order to let users "build up trusted object-centered networks, purely driven by their interests and their own quality considerations" [14, p. 4], connections between users in SRP are not necessarily synchronous, too. The concept utilizes "the fined grained XMPP-subscriptions" [14, p. 4] to "fine-tune the object-centered social networks" [14, p. 4], allowing one to "always choose to disable promotions from certain peers" [14, p. 4].

The fact that Google implemented asymmetrical connections one month after the author published a conceptual idea of it is a great confirmation for the concept and shows that the future of social networks might be based on asymmetrically connected peers. It also proves that the ideas are at the state of art.

7.5.2 Interoperability

While the author accused "the majority" of today's Web technologies not to be "connected and able to interoperate" [14, p. 1], this argument is only partially true when talking about Google+ and Google in general. With Takeout[37], Google promotes the honorable self-commitment (named Data Liberation Foundation[38]) that users have the "Freedom to leave" [22, p. 80] - and to take their data with them [22, p. 79]. To the present day [39], this means that parts of Google+ can be exported either in JavaScript Object Notation (JSON), vCard or HTML formats [22, p. 80]. All of them are open formats, which can be used to write importers for a certain service.

Exporting data in open formats is great, however, the author aimed his statement especially at live (in terms of "transparent") service interoperability, e.g. the transparent interchange of Facebook's Like and Google's +1. This is still not possible and will never be, as long as the services constitute "walled garden[s]" and thus violate against the very basic paradigms like standardization and decentralization of the Web [14, p. 1].

[34] `http://googleblog.blogspot.com/2011/06/introducing-google-project-real-life.html` (URL last access 2011-07-21).
[35] `https://plus.google.com/` (URL last access 2011-07-25).
[36] `https://www.google.com/intl/de/+/learnmore/index.html#circles` (URL last access 2011-07-25).
[37] `www.google.com/takeout` (URL last access 2011-07-25).
[38] `http://www.dataliberation.org` (URL last access 2011-07-25).
[39] 25th July 2011.

7.6 Service integration

Recent developments acknowledge the issue when working with different Web services - especially in unsteady environments with constantly new emerging services - each having it's own and special APIs: With Mozilla's Web Apps[40] or "Web of Apps" and Google's Web Intents[41], there are currently two approaches in abstractly describing a Web service in small JSON (Web Apps)[42] or Extensible Markup Language (XML) (Web Intents)[43] manifest files. Then, a consumer like a browser would, triggered by JavaScript, execute the activities as described, on the real services with their own API.

Both attempts are promising and could, once there exist a critical mass of implementations, solve the major connectivity issues Web developers are faced today. In addition, they could also boost the popularity of SRP implementations, as they would integrate seamlessly into established services without the sprawling integration efforts in wiring each search together by hand. Thus, Web Intents / - Apps can help liberating the Web service environment.

7.7 Related work

The idea of building a semantic agent based on XMPP is actually not new. The architecture of xOperator[44], for example, is related to the concept of this paper. However, xOperator focuses on executing (distributed) SPARQL queries via XMPP, while the approach in this paper is distributing the actual RDF-triples with it automatically.

Mozilla's RDF backend architecture features the inferface `nsIRDFObserver`[45] which can be used to monitor certain RDF statements.

Facebook's Like Button [46] implements a basic vote system as mentioned. However, the main difference is the underlying architecture: While Facebook can be considered to be a "walled garden" [10, p. 179], SRP is open, well defined, and decentralized.

[40]`https://apps.mozillalabs.com/` (URL last access 2011-08-24).

[41]`http://webintents.org/` (URL last access 2011-08-24).

[42]`https://developer.mozilla.org/en/OpenWebApps/The_Manifest` (URL last access 2011-08-24).

[43]`http://examples.webintents.org/` (URL last access 2011-08-24).

[44]`http://aksw.org/Projects/xOperator` (URL last access 2011-04-20).

[45]`http://www-archive.mozilla.org/rdf/back-end-architecture.html#Interfaces` (URL last access 2011-04-28).

[46]`http://www.facebook.com/help/?faq=13942` (URL last access 2011-04-28).

8 The prototype: SocIO

As announced in the position paper [14, p. 5], the author developed a prototype implementing the basic ideas of the SRP concept. The prototype is called SocIO and experiments with semantic-, Representational State Transfer (REST) and XMPP technologies in Java.

As mentioned on page 8, in regard of the Webinos project, mDNS and related, originally designed, local-network-aware protocols are not included in the prototype.

8.1 Scenario

As several comments suggested, the prototype concentrates on a very basic usage scenario of the SRP concept. SocIO thus supports the simple but expressive scenario described in the following:

SocIO provides a browser extension (most likely for Google Chrome[47]) as Graphical User Interface (GUI) - giving the user possibilities to tag the currently visited page and to view existing tags.

Tags are visually distinguishable between the user's own tags and the tags received from other users, and there are options to adopt respectively confirm foreign tags.

Without any special user interactions, all tags are transparently stored as RDF/N3 and exchanged with other agents in the background.

A suitable demonstration would be the pre-post comparison of tags to a certain Web page, before and after a peer contributed his tags with SRP.

8.2 Architecture

As figure 8 on the following page depicts, SocIO consists of two basic components to implement SRP, which are characterized as the following:

[47]Extensions in Google Chrome are basically local HTML files, enhanced with JavaScript and described by JSON. This makes the entire integration and use of SocIO quite transparent and is, in theory, also usable with other compliant browsers.

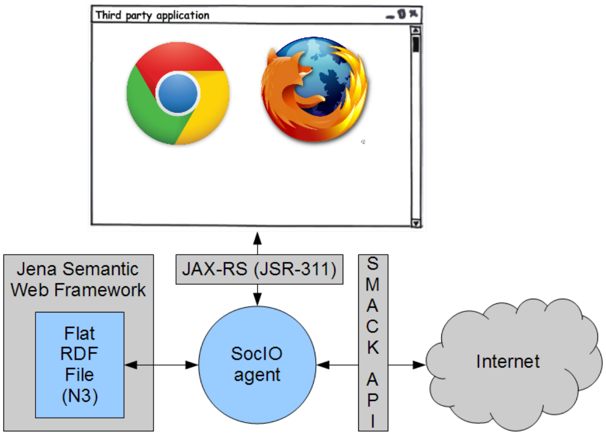

Figure 8: Architectural overview of SocIO: The three basic components are the XMPP-API Smack, the semantic core formed by the Jena RDF framework and a RESTful API based on JSR-311, to enable a simple browser integration.

1. Storage, query and exchange capacities with Semantic Web technologies.
2. Establish, sustain and manage connections to other SocIO agents with the XMPP protocol.

In addition, to support the basic browser integration scenario, SocIO has a third component which is not specified by the SRP concept:

3. Browser (plugin) communication via JavaScript and a RESTful API.

8.2.1 Semantic technologies

Jena[48] is the Semantic Web framework of choice in the prototype. It features RDF- and Web Ontology Language (OWL)-APIs, input and output functions for different notation formats like RDF/N3, persistence and query capabilities for SPARQL. In addition to the main project, there exist a number sub projects extending Jena with certain features, like the HTTP-interface for SPARQL-queries, Joseki[49], or the high performance storage subsystem TDB[50]. This modularity, the intuitive and powerful API as well as a comprehensive documentation[51] are notable advantages of the Jena Framework.

Joseki could be easily deployed to implement the query API, as mentioned in the concept on page 6. However, SocIO just uses Jena's native capacities, through the `SemanticCore` class for basic input/output and the `Semantics` class for wrapping capacities of the semantic model to Java.

8.2.2 XMPP client

The XMPP capabilities are provided by the open source client library Smack[52], maintained by the Ignite Realtime community[53]. Smacks makes it easy to send and receive messages via XMPP with a little effort.

By the time this document is written, SocIO broadcasts it's promotions to all users on the contact list (*roster*). However, Smack gives the option to access related XMPP properties, like groups or resource of a user[54], which could be utilized for selective broadcasts. In addition, there exists a Publish-Subscribe extension[55] for Smack, which could be used to implement event based notifications, as specified in XEP-0060[56].

Also, XMPP's subscription model, originally used in the concept to fine tune the connections within the users based on their interest ("object-centered social networks", see page 5), is basically supported by Smack[57] - but is currently unused by SocIO.

8.2.3 Browser integration

The concept of SRP intentionally does not specify a certain interface between applications and the SRP

[48] `http://jena.sourceforge.net/` (URL last access 2011-08-26).
[49] `http://www.joseki.org/` (URL last access 2011-08-26).
[50] `http://openjena.org/TDB/` (URL last access 2011-08-26).
[51] `http://jena.sourceforge.net/documentation.html` (URL last access 2011-08-26).
[52] `http://www.igniterealtime.org/projects/smack/` (URL last access 2011-08-28).
[53] `http://www.igniterealtime.org/` (URL last access 2011-08-28).
[54] `http://www.igniterealtime.org/builds/smack/docs/latest/javadoc/org/jivesoftware/smack/Roster.html` (URL last access 2011-08-28).
[55] `http://community.igniterealtime.org/message/191463#191463` (URL last access 2011-08-28).
[56] `http://xmpp.org/extensions/xep-0060.html` (URL last access 2011-08-28).
[57] `http://www.igniterealtime.org/builds/smack/docs/latest/documentation/roster.html` (URL last access 2011-08-28).

Listing 3: Contact initialization message (RDF/-Turtle format) sent by the SRP agent. [59]

```
@GET
@Path("/query")
@Produces(MediaType.APPLICATION_JSON)
public Response queryTag(@QueryParam("
    term") @DefaultValue("") String
    pattern);
```

agents. However, in order to communicate with the browser, SocIO features a REST API to enable easy integration of JavaScript via HTTP. Therefore, it uses Java API for RESTful Web Servics (JAX-RS), specified by JSR 311 and available since Java EE 6. The interface can be easily described by annotations, as figure 3 shows. SocIO's entire REST-API is described with the interface `SocIoRestApi`. The implementation class, `RestApi`, distributes calls to the respective Semantic or XMPP-component and maps the data format to JSON, e.g. for auto completion with JavaScript, if required.

However, because the script resource and the queried REST API are located at different locations, many current Web browsers would usually block direct communication, as this is the pattern usually exploited in Cross-site scripting (XSS) attacks[60]. The solution for this issue is Cross-Origin Resource Sharing (CORS), a W3C working draft[61]. With CORS, the server sends additional headers in this response, as figure 9 depicts. The respective headers are constructed by the `CorsResponse` class.

Figure 9: Two additional headers in the HTTP response of Jersey enable client-side cross-origin requests, specified by CORS.

[60] `http://en.wikipedia.org/wiki/Cross-site_scripting` (URL last access 2011-08-28).
[61] `http://www.w3.org/TR/cors/` (URL last access 2011-08-21).

8.3 Source code availability

SocIO is available[62] as open source on the social coding platform GitHub[63], bundled as Maven[64] project.

While SocIO itself is released as public domain respectively Creative Commons Zero (CC0)[65] license, it furthermore relies on open source libraries with compatible licenses like the Apache 2.0 license[66] and others[67].

8.4 Limitations

As mentioned, SocIO experiments with the basic technology stack of the SRP concept. In this context, security, performance, reliability and transactional stability are not subject of reflection. However, encountered, possible performance bottlenecks, especially the ones related with Jena, are notated with the Eclipse task tag `FIXME: This section is performance critical`.

As far as semantic technologies are concerned, there are limitations in the notation format (only RDF/N3 is supported) and, because of the lack of mDNS, there is no semantic contact initialization message (as described on page 6). Additionally, because there are no control messages at all, promotions cannot be re-requested again. Related to this, there is no way to query peers for certain promotions / resources.

Interests cannot be modelled, and therefore, there is currently no way to exchange promotions on a specific target group of users.

8.5 Conclusion to the prototype

SocIO is a basic but successful first implementation of the SRP concept. By focussing on a user friendly GUI and RDF storage and transport via XMPP, it combines all important technological building blocks of the concept with an expressive feature demonstration.

[62] `https://github.com/heussd/socio` (URL last access 2011-08-28).
[63] `https://github.com/` (URL last access 2011-08-28).
[64] `http://maven.apache.org/` (URL last access 2011-08-28).
[65] `http://creativecommons.org/publicdomain/zero/1.0/` (URL last access 2011-08-30).
[66] `http://www.apache.org/licenses/LICENSE-2.0.html` (URL last access 2011-08-30).
[67] For further information about the utilized third party libraries, see the Maven Project Object Model file, `pom.xml`.

9 Next steps

SRP is a powerful transport mechanism, featuring arbitrary usage scenarios and with SocIO, a promising Java implementation was realised. It can be stated that the standardisation of involved technologies, especially of XMPP and RDF, is usually sufficient - or at least in progress - for an implementation of SRP.

Therefore, SocIO is ready to be further developed and to be taken beyond its prototype status. Important starting points for that would be a further object centralisation, as mentioned, by utilizing more XMPP functionality, improvements in the RDF store regarding flexibility and performance, or the support for a native Java SRP-API.

The W3C Federated Social Web community will be informed about the state of the project. Also, integration scenarios into existing projects could be considered. For example, the news aggregator RSSOwl[68], which features a modular architecture and various extension points[69], could benefit from SRP as a first application.

10 Conclusion

The manner we collect, share and store folksonomies in the Social Web nowadays is interconnected with issues concerning interoperability and decentralization.

Aligned at the vision of a semantically enhanced Social Web, this paper proposes the concept Social Resource Promotion (SRP), which automatically maintains the entire end user-ecosystem of URIs. This includes the exchange of promotions for certain resources in form of a custom folksonomy within a certain group of peers. SRP is meant to reflect the web's dynamic and speed as well as promising developments, like being organized in object-centered social networks. Therefore, the author introduces an architecture consisting of usual Semantic Web infrastructure components like an RDF store and agent. In addition, the network stack features XMPP, to sustain communication via Internet, and mDNS to benefit from the fact that people with the same interests might meet physically, too.

This enables the automated creation of trusted, decentralized social networks of peers, sharing information of common interests. Thanks to the power of RDF, the kind of information content is not limited and can easily be customized, even if a first proposal uses the FOAF and the Tag ontology.

Central aspects of SRP's technology stack could be confirmed and further developed at the Federated Social Web Workshop 2011 in Berlin: The XMPP protocol, for example, used for transport and contact management, is deployed by a number of current projects for related reasons. An EU-founded research project uses mDNS for the detection of local devices. Also, a replacement for the utilized ontology was suggested.

In addition to the positive workshop experience, some developments happened even more recently, like Google's launch of Google+, implementing asymmetric social connections as independently proposed in the SRP concept. Or the recent approaches of Mozilla and Google in implementing service descriptions.

Encouraged by these developments, the author has prototyped SocIO, a first SRP implementation in Java. It consists of a JAX-RS-API, XMPP capacities by Smack and Jena Semantic Web Framework. Built of these building blocks, SocIO allows a basic tag and share scenario and can thus be seen as a successful prove of the SRP concept.

With SRP - and its first implementation SocIO - users have the option to create, maintain and share folksonomies in an open, well defined and decentralized manner. This is an important step towards the Social Semantic Web and allows everyone to create personalized, high-quality information pools with a maximum of diversity and in the very spirit of the Web.

The author welcomes contributors and anyone else interested in SocIO to collaborate on the social coding platform GitHub[70].

[68] `http://rssowl.org/` (URL last access 2011-08-29).

[69] `http://wiki.rssowl.org/index.php/Architecture_and_Extension_Points#Interesting_Extension_Points_provided_by_RSSOwl` (URL last access 2011-08-29).

[70] `https://github.com/heussd/socio` (URL last access 2011-08-31).

References

[1] webinos factsheet, June 2011. http://webinos.org/blog/2011/06/01/webinos-report-factsheet (last access 2011-08-10).

[2] webinos whitepaper, June 2011. http://webinos.org/wp-content/uploads/2011/06/webinos_whitepaper.pdf (last access 2011-08-11).

[3] Dean Allemang and James Hendler. *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2008.

[4] Iosif Alvertis, Michael Petychakis, George Gionis, and Robert Kleinfeld. A platform for managing campaigns over Social Media, 2001. W3C/PrimeLife Federated Social Web Europe Conference 2011 http://d-cent.org/fsw2011/wp-content/uploads/fsw2011-A-platform-for-managing-campaigns-over-Social-Media.pdf (URL last access 2011-07-27).

[5] T. Berners-Lee, R. Fielding, and L. Masinter. RFC 3986, Uniform Resource Identifier (URI): Generic Syntax, 2005. http://rfc.net/rfc3986.html (URL last access 2011-08-21).

[6] Tim Berners-Lee and Mark Fischetti. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. Harper San Francisco, 1st edition, 1999.

[7] Tim Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, May 2001. http://www.scientificamerican.com/article.cfm?id=the-semantic-web (URL last access 2011-08-15).

[8] Danah Boyd. About those walled gardens, 2007. http://www.zephoria.org/thoughts/archives/2007/02/05/about_those_wal.html (URL last access 2011-08-15).

[9] Herbert Braun. Facebook, Twitter und der Tod von RSS, 2011. http://heise.de/-1240619 (URL last access 2011-08-04).

[10] John G. Breslin, Alexandre Passant, and Stefan Decker. *The Social Semantic Web*. Springer, Berlin, 2009.

[11] Benjamin Carrillo and Julia Anaya. WebID-enabled XMPP: a bridge to the social web on your IM and beyond., 2011. Federated Social Web Europe Conference, June 3rd to 5th 2011, Berlin http://d-cent.org/fsw2011/wp-content/uploads/fsw2011-WebID-enabled-XMPP-a-bridge-to-the-social-web-on-your-IM-and-beyond.pdf (URL last access 2011-07-27).

[12] Diana Cheng and Daniel Appelquist. Privacy-aware XMPP-based microblogging: Lessons learned with OneSocialWeb, June 2011. http://d-cent.org/fsw2011/wp-content/uploads/OneSocialWeb-for-FSW.pdf (URL last access 2011-08-08).

[13] Thomas Gruber. Ontology of Folksonomy: A Mash-up of Apples and Oranges, 2007. http://tomgruber.org/writing/ontology-of-folksonomy.htm (URL last access 2011-08-29).

[14] Timm Heuss. Social Resource Promotion, 2011. Position Paper for Federated Social Web 2nd-5th May 2011n paper for the W3C Workshop on the Federated Social Web 2011 in Berlin, Germany http://d-cent.org/fsw2011/wp-content/uploads/fsw2011-Social-Resource-Promotion.pdf (URL last access 2011-06-15).

[15] Freddy Limpens, Alexandre Monnin, David Laniado, and Fabien Gandon. Nicetag ontology: tags as named graphs, September 2009. http://spin.nicta.org.au/SNI2009/SNI2009/Program_files/nicetag-full.pdf (URL last access 2011-08-10).

[16] Tim O'Reilly. What Is Web 2.0?, September 2005. http://www.oreilly.de/artikel/web20.html (URL last access 2011-08-01).

[17] Tim O'Reilly. Different Approaches to the Semantic Web, June 2007. http://radar.oreilly.com/archives/2007/03/different-appro-1.html (URL last access 2011-08-01).

[18] Emil Protalinski. Facebook blocks Google Chrome extension for exporting friends, 2011. http://www.zdnet.com/blog/facebook/facebook-blocks-google-chrome-extension-for-exporting-friends/1935 (URL last access 2011-07-28).

[19] Toby Segaran, Colin Evans, and Jamie Taylor. *Programming the Semantic Web*. O'Reilly, Beijing, 2009.

[20] Henry Story, Andrei Sambra, and Sebastian Tramp. Friending on the Social Web, 2011. Position Paper for Federated Social Web 2nd-5th May 2011, Berlin (Germany) http://d-cent.org/fsw2011/wp-content/uploads/fsw2011-Friending-On-The-Social-Web.pdf (URL last access 2011-07-27).

[21] Thomas Vander Wal. Folksonomy coinage and definition. Website, Februar 2007. http://vanderwal.net/folksonomy.html (URL last access 2011-08-28).

[22] Volker Weber. Google zieht Kreise. *c't - Magazin für Computertechnik*, 16(5):76–80, July 2011.