Bachelor Report

Developing a Browser Extension for a Social Bookmarking Service

by Florian Beyerlein,
Universität Bremen,
Fachbereich 3,
July, 2007

I hereby declare that this bachelor report was written by me. No other sources than those specified have been used.

Hiermit erkläre ich, dass dieser Bachelorreport von mir allein verfasst wurde und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet wurden.

Supervisors:
Prof. Dr. Andreas Breiter,
Prof. Dr. Rainer Malaka

Table of Contents

1 Introduction

According to the Encyclopædia Britannica an art collection is *"Works of art accumulated by an individual or institution"* (Encylopædia Britannica, 2007). They have been built since the *"earliest civilizations"* and consist of "*precious objects"* that *"were stored in temples, tombs, sanctuaries and palaces"* (Encylopædia Britannica, 2007).

Collecting valuable objects seems to be a very human propensity, and collectors have to pay attention to how to store them. Collectibles should have a special place where they are both saved, and can be shared with other people.

Internet users that find a notable World Wide Web (Web) page are faced with a similar problem: the webpage must be stored in a place where they can find it again, and where they are able to share it with others. The advantage of a webpage is that it can be collected by simply storing a reference to it, like a bookmark pointing to the URL[1] of the page.

The question for the user is in which 'tomb' or 'palace' to keep the reference?

There are two paradigms available from which a user can choose to save the page. One paradigm is to store the page locally on the user's computer, with the Web browser's bookmark management, or with a dedicated bookmarking application. The user has their collection saved and under their control, but this is also an isolated solution, where only they have access to it. Sharing requires extra activity by the user. The other paradigm is to save the URL to a Web-based storage system, which allows access from anywhere and, as it is already published on the Web, it can be easily shared with others (this is also called social bookmarking). Both ways are good and valid for their purposes; bookmarks a user wants to keep private can be stored on their computer, sites a user thinks are of public interest can be shared on a social bookmarking Web site. Apart from ubiquitous access to their bookmarks the user also gains access to human filtered Web pages with the shared bookmarks of others.

Taken together, this leaves two separate solutions to the user. When creating a bookmark they have to decide where to save it, to retrieve it they have to remember where it is saved, and when they decide to share a local bookmark they have to transfer it from one system to the other or vice versa, so that managing the collection can get complicated.

1 URL stands for Unified Resource Locator and is used here in the sense of an URI, a Unified Resource Identifier, which is the official standard for an address in the World Wide Web (see http://tools.ietf.org/html/rfc1630).

This short introduction demonstrates aspects of both bookmarking paradigms which are good for their purpose, but are also contradictory in the sense that using them together complicates collecting bookmarks. The question this work wants to answer is:

What can a system architecture look like to enable the privacy needs of users, while enabling social bookmarking?

For that reason, an analysis of both local bookmarking and social bookmarking is undertaken to better understand the user's issues. As a third topic, the issue of privacy emerges while sharing personal items on the Web, and must be illuminated. Based on the findings in these three domains, a system called *Social Bookmark* is constructed, that integrates both local and social bookmarking with regard to the user's privacy needs. Finally, a prototype implementation is presented that shows the feasibility of *Social Bookmark.*

2 Personal Bookmarking

What is bookmarking? According to the paper "Information Archiving with Bookmarks: Personal Web Space Construction and Organization" (Abrams, Baecker & Chignell, 1998) it is *"dealing with information overload"* (Abrams, Baecker & Chignell, 1998 p. 41) and the answer to the experience called *"lost in hyperspace"* (Nielsen, 1990 p. 127) caused by the absence of a standardized structure of the Web. The lack of structure in the Web urges the users to *"develop personal information systems",* like bookmarks, which are utilized by the users to build a *"personal Web information space"* (Abrams, Baecker & Chignell, 1998 p. 41). But bookmarking is not only the creation of bookmarks but *"[...] structuring the resulting collection, and managing it's growth"* (Abrams, Baecker & Chignell, 1998 p. 41) and also a means of personal information management as it is the "*collection, storage, organization and retrieval of digital objects"* (Boardman & Sasse, 2004).

2.1 Why users bookmark

According to a survey of the Graphic, Visualization, & Usability Center at the Georgia Institute of Technology from 1998 on World Wide Web usage shows that most users employ bookmarks to some extent (only 3.5% do not know or do not use bookmarks) and 34.9% of the users have more than 100 bookmarks (Pitkow, 1998)[2].

The users' criteria to create a bookmark for a page depend on their personal judgment of the pages' value and this value is changing over time (Pirolli, 1995).

Abrams found five distinct reasons in his thesis *"Human Factors of Personal Information Spaces"* why users bookmark a webpage: *'general usefulness'*, *'quality'*, *'personal interest'*, *'frequency of use'* and *'potential future use'* (Abrams, 1997 chapter 5-11).

General usefulness refers to the user's personal perception of a page's utility. This perception changes over time and thus the criterion gains importance again when bookmarks are reorganized at a later date. *Quality* attributes to a Web page's function to act as a holder of references to other 'important' Web pages. *Personal interest* is a reason to use bookmarks as shortcuts to interesting pages, too. This criterion also means, that bookmark collections represent at least a part of a person's interests (see also Boardman & Sasse, 2004 p. 587). To reduce the effort to look up the same information source repeatedly, bookmarks are created because of a high *frequency of usage*. At last, bookmarks are created to be retrieved and (re-)evaluated in a *future*

2 URL of the survey result on "Number of Items in Favorites/Bookmark": http://www-static.cc.gatech.edu/gvu/user_surveys/survey-1998-10/graphs/use/q62.htm.

browser session.

2.2 How users use bookmarks

In his thesis Abrams also proposed a *"Taxonomy of bookmark use"* with the three basic nodes *'Reducing the user load'*, *'Facilitating navigation/access'* and *'Collaborating/publishing/archiving'* (Abrams, 1997 chapter 5-13 – 5-22). *'Reducing the user load'* means bookmarks help in managing URLs and are used as *"memory aid"* as well as a history mechanism to the user. Bookmarks help *'Facilitating navigation/access'* by accelerating the access to and help in finding information. Further bookmarking is used to "*create their own personal information space and share it with others"* (Abrams, 1997 chapter 5-20).

2.3 How users manage bookmarks

The following paragraphs summarize the findings of the thesis on bookmark management (Abrams, 1997 chapter 6). These findings are based on 'state of the art' bookmark management in 1998. Most users of the study used the browser built-in bookmark management (over 95%) to some extent. It could be argued that this study is outdated, but even today the both flavors of mainstream browsers[3] Microsoft Internet Explorer and Mozilla Firefox use basically the same organizational paradigm: bookmarks are created and filed in a (possibly) hierarchical structure which can be built up and managed by the users to their need. The retrieval of bookmarks is done by walking the bookmark tree and listing of bookmarks in a trees node.

2.3.1 Organization of bookmarks

Abram defines seven methods for organizing bookmarks (*'No Organization', 'Ordered List', 'Set', 'Hierarchy', 'External', 'Other')* (Abrams, 1997 chapter 6-2) and found that the number of people who didn't organize their bookmarks (total of 36.9%) decreased steadily with a growing number of bookmarks (Abrams, 1997 chapter 6-3). He stated "*three primary factors: expertise, time/effort, and number of bookmarks"* (Abrams, 1997 chapter 6-3) as reasons not to organize bookmarks. Some people just don't know how to organize their bookmarks, the effort to organize bookmarks is too high in consideration of the benefit or there is just no need to organize bookmarks due to their small number. Ordered lists had a low percentage of users only (about 10%)

3 This statement is made based on random browser statistics from different providers, which show an accumulated percentage of over 80% for all versions of Microsoft Internet Explorer and Mozilla (Firefox); Sources used are: Upsdell, 2007; TheCounter, 2007; W3Schools, 2007; NetApplications, 2007.

(Abrams, 1997 chapter 6-4) and, as the number of bookmarks grows, more organized methods were used. In the group of people with more than 26 bookmarks over 50% used a set or hierarchy of bookmarks to organize and a high percentage (44.4%) used hierarchies when they had over 300 bookmarks (Abrams, 1997 chapter 6-5 – 6-6). Only few users (2.2%) chose to publish bookmarks on a webpage to gain flexibility in organizing and annotating them (Abrams, 1997 chapter 6-6). 44,4% of people with more than 300 bookmarks use external programs (Abrams, 1997 chapter 6-7). As reasons to manage bookmarks externally is stated: "Existing tools are difficult to use, they are not powerful enough to manage a large number of bookmarks, and they have a steep learning curve" (Abrams, 1997 chapter 6-7).

For the frequency of bookmark organization Abrams gives four groups, 'No filers', 'Creation-time filers', End-of-session filers' and 'Sporadic filers' (see Illustration 1). The strategy chosen by users depends on the benefits they gain weighted by the effort for the strategy. About half of the users perform sporadic filing (when the need to organize is high) and only 7% do end-of-session filing. This is explained by the fact that most users only create up to 5 bookmarks per session which is a number to scope

Frequency of Organizing Bookmarks

No Filers: users who never organize bookmarks. The bookmarks stay in the order in which they were created.

Creation-time Filers: users who store a new bookmark in the appropriate category when the bookmark is first created.

End-of-session Filers: users who organize all their new bookmarks at the end of the session.

Sporadic Filers: users who organize bookmarks occasionally or sporadically. Sporadic Filers may go many sessions without organizing bookmarks.

after: (Abrams, 1997 chapter 6-9)

Illustration 1: Frequency of Organizing Bookmarks

with unfiled. Interestingly 67% of the users with more than 300 bookmarks file at creation time. "*For these users the benefits of an up-to-date organizational structure far outweighs the distraction from browsing that is required to file a bookmark*" (Abrams, 1997 chapter 6-12). The scale factor is important to the organization of bookmarks (Abrams, 1997 chapter 6-20). Users of the study complained about the difficulties of dealing with large number of bookmarks and some started to employ different programs to organize the bookmarks since their current method lacked options. In general Abrams says "*large scale requires more organization*" (Abrams, 1997 chapter 6-21).

2.4 Structuring bookmark collections

The structure of bookmark collections is both necessity and a challenge for the user. As said higher scales of bookmarks require more structure, the users build their own semantic hierarchy and use it to navigate their personal information space (Abrams, 1997 chapter 7-2). The challenges for the user lay on both sides. Building a hierarchy requires *"cognitive effort and time"* and *"domain expertise"* (Abrams, 1997 chapter 7-3). Importance lays in the stability of the semantic hierarchy to facilitate filing and retrieval by making the cognitive effort to memorize the structure as low as possible. Stability and consistency of the semantic is on the other hand difficult to achieve *"because the significance of categories changes over time"* (Abrams, 1997 chapter 7-5).

2.5 Complexity problems

As the classification of an information space ranges continuously from a simple, personal to a complex one, bookmark collections can grow into complex information spaces if the users do not manage to keep them simple (Abrams, 1997 chapter 9-1). Problems that can arise are *'Information overload'* when the valuable information can not be handled anymore (Abrams, 1997 chapter 9-1), *'Pollution'* when bookmarks are created that are not used (anymore) (Abrams, 1997 chapter 9-2), *'Entropy'* as effort to keep the complexity of the collection low and problems of visualizing and structuring larger numbers of bookmarks (Abrams, 1997 chapter 9-3).

2.6 Summary

Bookmarking is a means to deal with the information overload that is presented by the world wide web. Users try to reduce this overload by building an information space they can handle. As the information space grows over time its management becomes more and more important. The hierarchical structure of bookmark collections, which is the main paradigm in browsers (even today), presents a non scalable way to organize bookmarks manually.

3 Social Bookmarking

As the above stated issues are known for a long time different approaches were made to address both the handling of the complex information space of the Internet and the handling of personal information spaces. One strategy is the use of collaboration to solve problems of "*resource discovery*", "*recall*" and "*maintenance*" (Kanawati & Malek, 2000). A number of systems have been developed to make use of collaboration like *CoWing* (Kanawati & Malek, 2002), a multi-agent system by (El-Beltagy, DeRoure & Hall, 1999) or *WebTagger* (Keller et al, 1997). These systems employ additional software (software agents and proxies) to assist the users in managing their online information.

In the recent years another collaborative approach has become a solution with a broader user base: social bookmarking or public link management (Hammond et al, 2005). Hammond et al. see it both as server-side software for managing bookmarks and the unstructured approach of tagging for organization (Hammond et al, 2005). This server-side approach gives users a nice side-effect, that probably also has an influence on the popularity of these applications: the user has access to their bookmarks from every Web accessible computer.

The following chapter displays social bookmarking in the context of social software, its main organizational paradigm called '*tagging*' and its solution to the resource discovery by applying the social navigation technique of collaborative filtering.

3.1 Social Software

The term social software according to Richter & Koch is used for software that uses *AJAX*[4] to create Web sites with desktop application like user interfaces or *RSS*[5] to make content of Web sites available to other Web sites (Richter & Koch, 2007 p. 4). But apart from technological characteristics the active involvement of users as content creators in new forms of online applications like "*Weblogs, Wikis, Mashups, Social Bookmarking, ...*" (Richter & Koch, 2007 p. 5) are the formative features.

4 Asynchronous JavaScript and XML, see Wikipedia, 2007a.

5 Really Simple Syndication (RSS 2.0), RDF Site Summary (RSS 1.0 and RSS 0.90) or Rich Site Summary (RSS 0.91) are XML based documents that allow publication of contents in a standardized manner, see Wikipedia, 2007b.

Schmidt (2006) proposes the following definition:

> *"Social Software refers to those online-based applications and services that facilitate information management, identity management, and relationship management by providing (partial) publics of hypertextual and social networks."*

Though the term management leads to recognizing the activities as consciously planned, they are more seen as a result of "[...] *apply[ing] certain communicative strategies [...] which might be implicit and not reflected at all"* (Schmidt, 2006 p. 33). He also argues that the separation of the three management types is artificial and in practice always a mixture will be found.

A social bookmarking service (SBS)[6], for example, is mainly used for collaborative information management, but can feature relationship management through maintaining or helping to create contact with users of similar interest.

3.2 Tagging and Folksonomies

One crucial factor in personal bookmarking, as described in chapter 2 Personal Bookmarking, is the creation of a semantic hierarchy in which a particular bookmark is placed. Alternatively user generated attributes in form of keywords or tags can be assigned to resources, commonly called tagging. When tagging is done by many people on shared resources, it is also called *collaborative tagging* (Golder & Huberman, 2006) or better known under the catchy phrase *folksonomy* (Sterling, 2005). The option to freely assign tags to resources meets the user's need to ease the categorization, because they do not have to place a bookmark in one particular path[7] of a hierarchy but can choose keywords which seem now[8] appropriate to them for the given resource (Bielenberg & Zacher, 2005 p. 13f).

The retrieval of bookmarks basically happens through filtering the collection by one or more tags. How the filter tags are provided by the user is manifold; two most common ways are: a text input field in which the tags are typed and tag-clouds where a user chooses a tag by clicking on it. Tag-clouds are lists of tags with different sizes and colors reflecting some of their attributes like their distribution in the whole folksonomy or some ranking value.

When the individual tags are being used by many, both for searching and categorizing, opportunities and issues arise. The opportunities are ways of generating meta-data from the collaboratively compiled data. Examples for this are

6 Like http://del.icio.us, http://furl.net or http://mister-wong.de.
7 It is takes more effort to decide on each branch of a hierarchy which way to go than simply to state words that come to the user's mind.
8 The time of creation.

the support of tagging and filtering (Hassan-Montero & Herrero-Solana, 2006; Halvey & Keane, 2007) or the creation of groups from tagging data (Bielenberg & Zacher, 2005; Jäschke et al, 2006).

Problematic issues emerge because "*tags may be interpreted differently outside of their creator's context*" (Bielenberg & Zacher, 2005 p. 15). Richter & Koch describe several cases of problems: tags with the same meaning but different spelling (foto, photo), singular/plural forms of words (song/songs), different languages (house/haus), synonyms (television/TV), or homonymy (bank) (Richter & Koch, 2007 p. 24). Also the expertise of the tagger plays a role when tagging a resource; the more expertise the tagger has the more specific are the tags that can be given (Richter & Koch, 2007 p. 24). Despite the freedom of each user to tag resources to their liking research has shown that the tags used for particular resources stabilize in their variation and it is no problem for tagging systems and to allow individual use of tags (Golder & Huberman, 2006). This stable pattern of collectively assigned tags is explained with *"imitation"* of tag usage[9] and a *"shared knowledge"* which leads to shared naming schemes for objects (Golder & Huberman, 2006).

3.3 Social Navigation

Tagging and the resulting folksonomies are a form of social navigation; for example Dieberger (Dieberger, 2003 p. 294) states:

"A goal of social navigation is to utilise information about other people's behaviour for our own navigational decisions."

As examples he states the number of cars in front of a restaurant as a hint about its popularity or a recommendation for a doctor from friends (Dieberger, 2003 p. 294). Both show how other peoples' behavior or experience can influence a personal decision. The first is an indirect observation of what other people have done and the observer can decide if it is a good sign to choose a popular place to eat or not. The second is a direct, personal recommendation and requires an active recommender. Social bookmarking websites use both approaches: direct by giving the possibility to hint a bookmarked URL to other users[10] and indirect for example by providing lists of popular URLs[11] or by searching in others bookmarks by tag. When searching in this form, a user harvests the effort of other people to describe a URL with tags.

9 For example popular tags by other users for a URL are shown on the bookmarking service del.icio.us when creating a new bookmark (del.icio.us, 2007a).

10 On del.icio.us the "network" functionality.

11 Popular Web sites, that is URLs tagged by many people, are listed for example at http://del.icio.us/popular.

3.4 Summary

As has been shown social bookmarking is a way to meet the individual problems of bookmarking and bringing up new opportunities for the whole of the Internet users. The organization paradigm 'tagging' eases the individual effort of bookmark management and the result of collaborative tagging is used both to support individual bookmarking and for filtering relevant URLs from the Internet. As the collaborative bookmarking services gain usefulness through the number of active participants they are freely accessible. The services looked at as a preparation to the implementation of the proposed *Social Bookmark* system only required an email address and a password (see 7.1 Choosing a social bookmarking service).

4 Privacy on the Web

When visitors of Web sites provide data, privacy is gaining importance. To what privacy means Belloti & Sellen (1993 p. 78) say:

> *Any realistic definition of privacy cannot be static. With the introduction of new technology, patterns of use and social norms develop around it and what is deemed "acceptable" behaviour is subject to change.*

and they further define *"[...] privacy to be a personal notion shaped by culturally determined expectations and perceptions about one's environment."* (Belloti & Sellen, 1993 p. 78). Their point is that *"Technology is not neutral"* towards users privacy as it can enhance or reduce peoples' *"control over personal data"* (Belloti & Sellen, 1993 p. 78).

Why this control over data is so important lies in the nature of privacy breaches; once data has left the control of a person, there is no way to re-establish it again (Martin et al, 2000 p. 5).

The insight that network systems, where users interact or communicate with each other, need privacy mechanisms are not new and scientific approaches to build collaboration systems (Rotenberg, 1993; Kanawati & Malek, 2002; Dieberger, 2003) take this into account. Dieberger argues that *"Care must be taken that a system cannot reveal private information about a user"*, at least not when the user is not aware of the revelation (Dieberger, 2003 p. 302).

Forsberg, Höök & Svensson (1998) argue in their paper *"Design Principles for Social Navigation Tools*" amongst other criteria that trust and privacy are of importance. Trust must be present for a user to be certain that the information they use from someone else is reliable and, that they should know what the others intentions are. As for privacy they argue for a *"discretion policy"* of the Web site about information a user leaves (Forsberg, Höök & Svensson, 1998).

These privacy policies state the services' intention how to deal with provided data, but this is not a guarantee[12]. Centralized architectures, as Web services are, have *"single point of failures problems"*, like system availability, responsiveness, scaling and trust (Kanawati & Malek, 2000). Foner especially sees trust as a problem of centralized systems (Foner, 1999). As social bookmarking services are centralized, they can not necessarily be trusted.

Kanawati & Malek (2000) argue that users should have the possibility to control to whom their information is revealed and this control can not be guaranteed by a

12 An incident on the flickr (http://flickr.com) online photo sharing service (Donath, 2007) shows computer failures can expose those privately declared items.

centralized solution.

As an interesting notion the aforementioned privacy policies are only read by a marginal number of people (Kohavi, 2001). This behavior could be explained with the users perception of anonymity in the World Wide Web but *"The Internet is not a safe haven for sensitive information"* (Dieberger, 2003 p. 302).

But even if everything is in order, i.e. the Web services are reachable, they handle the given data carefully, and only the data the users declare as public are public, there is the risk of re-identification. Re-identification is the linking of anonymous to identifying data, to achieve a gain in the information known about the identified person. Some studies have been conducted that show how re-identification can take place through public statements of users. For example Gross & Acquisti, (2005) analyze the social networking website 'facebook'[13] according to information revelation of the users and the risks the users expose themselves. They see the possibility of stalking, the building of a digital dossier of a person and re-identification via facial[14] and demographics re-identification. (Frankowski et al, 2006) used 'sparse relation spaces'[15] to re-identify people from the movie recommender service 'movielens'[16]. One could argue that the information revealed by social networking sites are very different from social bookmarking services. The first rely on personal data that is made public, the second only needs pseudonymity[17]. But a user's bookmark collection resemble to a sparse relation space which could be combined with other sparse relation spaces for re-identification.

13 See http://www.facebook.com.

14 Facial re-identification was achieved by matching profile images from different online services

15 *"A sparse relation space is a dataset that (a) relates people to items; (b) is sparse, having relatively few relationships recorded per person; and (c) involves a relatively large space of items."* (Frankowski et al, 2006)

16 See http://movielens.umn.edu.

17 Pseudonymity means the obfuscation of an identity by using a pseudonym.

5 Summary

This chapter will summarize the analysis of the chapters 2 Personal Bookmarking, 3 Social Bookmarking and 4 Privacy on the Web. This is taken as a basis for a proposal of a software system that combines in-browser personal bookmarking with social bookmarking to address privacy.

Bookmarking is done by people to set marks in the vast information space of the Internet. These marks serve them as hints where they have been in the space, perhaps to come back, or as references and shortcuts where to go next time they explore a certain trail they did not the last time.

As the act of marking takes some effort it makes sense to mark locations of value and interest. The resulting user's collection of bookmarks thus have some value to the users and reflect (at least some of) their interests.

A collection of bookmarks must be managed to stay valuable to the user. Depending on the number of bookmarks users have they pursue different kinds of organizational structuring. People with a small number of bookmarks gain nothing from bookmark management, whereas with increasing numbers of bookmarks management is needed to keep the value of a bookmark collection, otherwise it turns into a vast information space. One traditional way of organizing larger information spaces (like a library) is the building of a semantic hierarchy, which is done by professional staff that has the knowledge to do so, and place the items in question (books) in the hierarchy (categorize) (Rowley, 1992). Widely used browser software uses this pattern to organize bookmarks of Web pages and people use it for their collections. But average users are not professional knowledge organizers and probably have difficulties to build and maintain their personal taxonomy.

This offline, personal form of bookmark management is rivaled by an online social, collaborative approach, where people share their bookmarks in a centralized manner on a Web site. Social bookmarking services use tagging as an easy to use and maintain organizational pattern. The combination of individual tagging results in folksonomies, which are a tool for users to explore and search the personal and public information space of the social bookmarking service.

As a drawback of social bookmarking services their users abandon part of their privacy. Bookmarks are publicly attributable to certain pseudonyms so users must be alert, that they do not unintentionally expose information they rather kept private. Technological systems should have mechanisms that prevent unwanted privacy breaches.

6 A Personal/Social Bookmarking Architecture: SocialBookmark

Taken the privacy concerns as basis this chapter lays out a software system called *Social Bookmark* that facilitates the personal bookmarking needs and supports social bookmarking as well.

6.1 Privacy Enabled Social Bookmarking

The basic concept is to let the user stay in control of their data, both logically and physically. This means the bookmarking data must be saved on the user's side. In this case two basic approaches are possible: Either the data is stored and managed in a separate application[18] or within the browser application. In *Social Bookmark* the bookmarking data is stored within the browser. The advantage that is seen over a separate application from the users perspective is a more streamlined workflow. This means that the user does not have to switch the application context during a Web browsing session, bookmarking and bookmark retrieval can be done with little interruption.

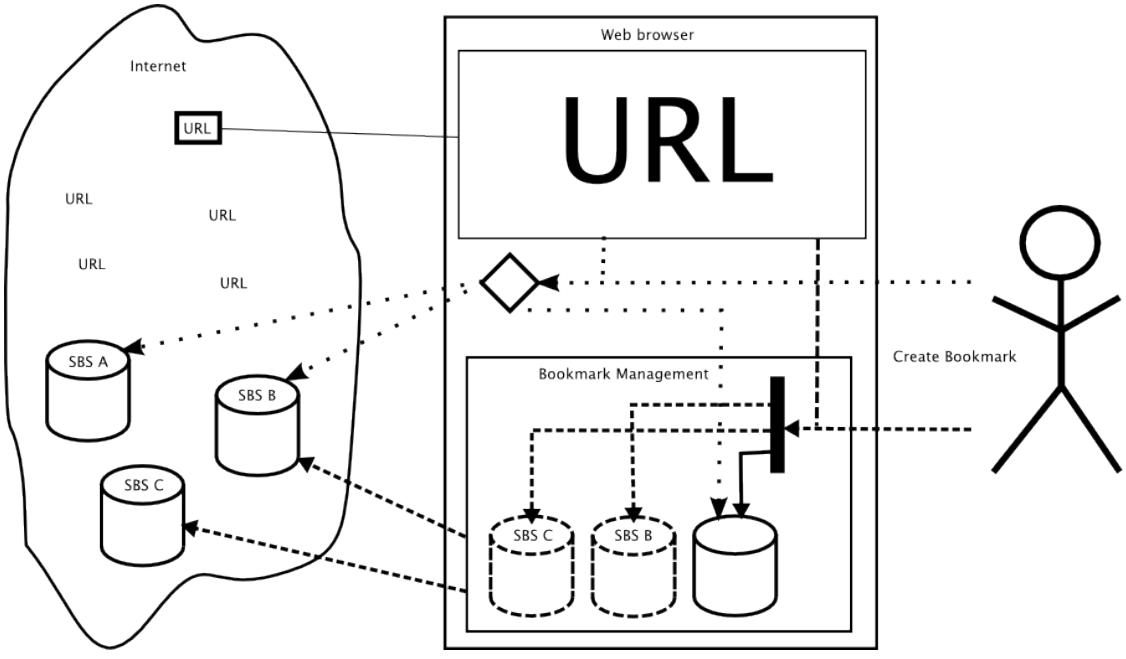

Illustration 2: Creating a bookmark; without (dotted) and with SocialBookmark (dashed)

18 A 'separate application' means an additional software to the Web browser.

When the user creates a bookmark for a URL they have the option to declare the bookmark public or private depending on their decision if it should be available for the public or if it is (yet) too personal to give away. The system architecture allows for an even more fine-grained privacy status: A bookmark can be local private, then it is not even sent to the online service, or it can be just private, in which case it is saved online but with a privacy marker, if supported by the service.

6.2 Multiple Services

According to Abrams (Abrams, 1997 chapter 5-20) people use bookmarks for different reasons and correspondingly there are different social bookmarking services that all capture the users' bookmarks, but have a different focus on how the bookmarks are utilized (see 7.1 Choosing a social bookmarking service). So people can have the desire to make bookmarks on different SBS, on the other hand from the user's perspective all bookmarks are of some value to them so they should be kept in one place. The Social Bookmark system must therefore present the user with the option to publish a bookmark to one or more social bookmarking services.

As a prerequisite for a social bookmarking service to be supportable the service must provide an Application Programming Interface (API) that enables third party applications to use basic functionality; in this case support for saving, editing and deletion of a bookmark must be present as well as retrieval of the users bookmarks from the remote service for synchronizing purposes.

The support for several social bookmark services is achieved by a plug-in mechanism. For each service that should be supported a plug-in has to be

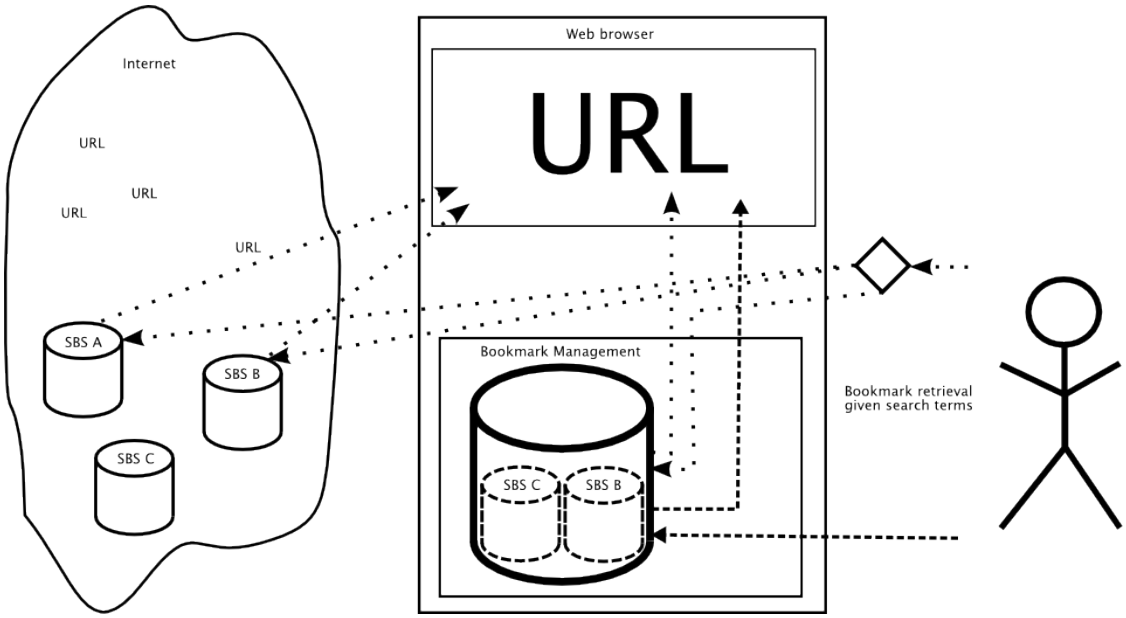

Illustration 3: Retrieving a bookmark; without (dotted) and with SocialBookmark (dashed)

programmed that implements a given application interface of the *Social Bookmark* software. This interface provides functions for saving (editing) and deleting bookmarks.

This option has also a privacy side effect: when used to bookmark URLs generally to more than one public service the risk of re-identification arises, because in this case the user creates several sparse relation spaces from which re-identification can occur. On the other hand more sparse relation spaces exist that must be analyzed to re-identify, which raises the effort to do so.

6.3 Bookmark Management

Social Bookmark utilizes the tagging pattern to organize the user's bookmarks, for two reasons: (1) it is easy to use and (2) it facilitates the integration in existing social software services.

1. Easy to use

The creation of bookmarks is leveraged (in comparison to the hierarchy pattern) as a bookmark is correctly categorized as soon as a tag is assigned. Reorganization too is facilitated by tagging; to change or refine a bookmarks' classification, only single tags have to be changed or added.

2. Adoption of social bookmarking service pattern

As tagging is the most used organizational pattern of social bookmark services[19] it would be more effort to convert between a hierarchical pattern and a flat namespace.

Additional to the tagging pattern users must be able to alter any attributes of the bookmark including privacy settings (local private/private/public).

6.4 Synchronization

Having the option to create and store bookmarks at several places, inside the browser or at one of the several online SBS an new issue must be targeted: consistency of the user's whole bookmark collection through synchronization. Two cases can be imagined, these are offline bookmarking in *Social Bookmark* and online bookmarking circumventing the *Social Bookmark* software. In both cases an option must be present to bring both storages to hold the same information, newer and updated information from each online storage must be read and stored in the *Social Bookmark*s offline storage and bookmarks that have been altered in *Social Bookmark* software while not being connected to the Internet must be saved to the selected online storages. This feature also allows the user to use the *Social Bookmark* software

19 All but one (http://www.stumbleupon.com) from (Iskold, 2006) use tagging.

at different computers with the drawback that their local private bookmarks from one installation cannot be transferred to an other instance of *Social Bookmark*. This case is part of further research and not covered in this work.

7 Prototype of Concept: Social Bookmark

Having outlined the *Social Bookmark* concept the following chapter shows an example implementation of the concept to test its feasibility.

One requirement of the *Social Bookmark* concept is that it integrates in the browser to allow for a streamlined workflow (see 6.1 Privacy Enabled Social Bookmarking). This raises the question how a browser application can be modified by external software that allows such integration. Mozilla Firefox (Mozilla, 2007) was chosen as the target browser because it is

1. a well-established and widely-used browser
2. it is open source, which means that every aspect of it can be modified,
3. it has an mechanism that allows the development of third party extensions (see Mozilla.org, 2007a),
4. the development of the user interface is done in XUL (which is XML),
5. the programming of the logic is done in JavaScript,
6. Firefox Version 2 has a build-in SQLite database (SQLite, 2007a), that can be used to build the storage system that is needed.

The last 3 reasons are important for the feasibility test, because the author already has experience in these technologies used for extension building.

One important decision has been made regarding the implementation: in this step only one social bookmarking service should be supported. This decision reduces the complexity of the plug-in enormously as questions of managing several plug-ins drop out. This influences the user interface (where must what functionality for which services be present) as well as the management of the bookmark data itself (to which services does a bookmark belong to? what additional data for a particular service is attached to a bookmark? how are bookmarks synchronized?). It must be said that this decision does not affect the key principle of *Social Bookmark*: that the user's data should stay in the user's realm and that they can share it from within that system.

As a preparation to the implementation two small scale analyses were conducted. One to choose an existing social bookmark service that is the basis for implementation and the next to define the actual functionality of the Social Bookmark software.

7.1 Choosing a social bookmarking service

This chapter shows a small evaluation of a few existing social bookmarking services. First the evaluation criteria are defined, then the criteria are applied to a selection of social bookmarking services.

The criteria for evaluation are the following:

1. Pseudonymity / Account data

How much does the person have to disclose of itself? This criterion is important, because it shows how easy it is for a person to start using a service. If it is only a nickname, password and an E-mail address (which is common to facilitate password retrieval and hinder spam bots), then it is called *pseudonymity*, if personal data like, country, year of birth or real name are asked it is a form of *personal identification.*

2. Tagging

Does the service use tagging as the main organizational paradigm for the bookmark collection? This is important as *Social Bookmark* too aims at using tagging as bookmark organization method.

3. Private bookmarks

Does the SBS allow private bookmarks or are all individual bookmarks public? This feature is interesting for the user as it allows in combination with Social Bookmark a more fine-grained release method.

4. API

Social Bookmark relies on an SBS' openness in terms of an Application Programming Interface (API) that allows saving, editing, deleting and retrieving of user's bookmarks. This requirement is essential.

The services have been selected for this evaluation from an incomprehensive list of existing services (Iskold, 2006). The three most popular services (according to the Alexa[20] und Technorati[21] ratings in Iskold, 2006) have been selected, these are del.icio.us (del.icio.us, 2007b), stumbleupon.com (stumbleupon, 2007) and furl.net (furl.net, 2007).

Service \ Criteria	Pseudonymity	Tagging	Private bookmarks	API
del.icio.us	Yes	Yes	Yes	Yes
stumbleupon.com	Yes	Yes	No	No
furl.net	Yes	Yes	Yes	No

Illustration 4: Evaluation results for chosen SBS

20 See http://www.alexa.com.
21 See http://technorati.com.

furl.net (furl.net, 2007).

The result of the evaluation is summarized in Illustration 4. The order of the services is by popularity according to Alexa and Technorati.

7.1.1 Pseudonymity

del.icio.us and furl.net require only non-personal statements prior to using them with the small exception of furl.net which requires to state that one is over 13 yeas of age. Nevertheless this can be neglected from a privacy perspective as it does not tell much. Stumbleupon.com requires birthday and gender to be stated which can lead to re-identification, but this is not necessary for the use of bookmarking functionality so a false statement does not prevent the services use.

7.1.2 Tagging

All of the services provide tagging functionality. At del.icio.us and furl.net tagging is an essential part of creating a bookmark, the tagging functionality of stumbleupon.com is existing but hidden, because the focus of stumbleupon.com lays more in building social networks through browsing behavior than as a repository for bookmarks.

7.1.3 Private Bookmarking

del.icio.us and furl.net allow users to declare a bookmark private, whereupon the bookmark is hidden from the public. Again for the different intention of stumbleupon.com there is no privacy setting for bookmarks, all bookmarks a user saves can be assigned publicly to the user.

7.1.4 API

The only service that provides an API for third party developers is del.icio.us. The API provides bookmarking functionality and both access to the users bookmarks and tags via secure HTTP (https, see Wikipedia, 2007c)

7.1.5 Summary

From this small selection of popular social bookmarking services only one fulfills all criteria required to implement the *Social Bookmark* architecture: del.icio.us. The critical criterion is the existence of an API that third party developers can use to address the service. Del.icio.us' primary use is the building and management of a personal bookmark collection which suites the intention of *Social Bookmark* as an architecture for personal bookmarking needs.

7.2 Analysis of SBS (del.icio.us) functionality

As the concept of Social Bookmark should be developed only for the del.icio.us SBS to prove its feasibility an analysis of the del.icio.us website functionality is carried out. In particular bookmarking and searching for bookmarks on the website are examined. The networking functionality of del.icio.us is disregarded, as networking is not the primary goal of *Social Bookmark*. Finally the provided API is evaluated in terms of its functionality. This analysis provides the information needed for implementation apart from the functionality that is derived from the *Social Bookmark* concept.

7.2.1 Bookmarking

The most important function is setting a bookmark. There are several ways to create a bookmark on del.icio.us, either the webpage a user wants to bookmark has a link backed by a JavaScript function that opens a pop-up window with a form to set the entries, or the user has clicked on a *bookmarklet*[22] which calls the bookmarking page of the del.icio.us service. In both cases some fields of the bookmarking form are pre-filled (the URL, title of the page and optionally the users selected text of the webpage as a note) but the tags field are empty leaving the user to write as many tags as wanted to describe the page. The tagging is supported by up to 4 lists of tags that are proposed to describe the page:

1. “Recommended tags”, that have been used for the page by others and used by the user on any other page. 2. All the users own tags (“your tags”). 3. Special “your network”[23] tags (“for:XXX”), which are used to hint this page to another user (XXX) that is in the “network” of the posting user. 4. “popular tags” which are used by other people for this page.

The words in these lists can be clicked to add them to (or remove them from) the tag form field or can be auto-completed by writing the beginning of a tag, selecting the completion with the up- and down- arrows and hitting tab.

22 A *bookmarklet* is a bookmark that targets a JavaScript function instead of an URL.

23 A user on del.icio.us can decide that another user should be in their “network”. Then they have fast access to the user's bookmarks by filtering for the other's username.

Illustration 5: Bookmarking a URL on del.icio.us in a pop-up window

A bookmark can be declared private by checking the "do not share" check-box beside the "url" input field.

With the "save" button the form is submitted to del.icio.us and the window closes. The user has saved a new bookmark with annotations to their bookmarks.

There are two other ways to create a bookmark to the del.icio.us service. One by setting all fields manually and one by bookmarking a search result (or a page from the recent post/popular posts list) from within del.icio.us.

7.2.2 Searching Bookmarks

Searching for bookmarks can be initiated in several ways. Within the del.icio.us website it is possible to click on a tag and get a list of websites that are annotated with this tag; additionally a search box exists where any word or word combination can be submitted as tags/search terms.

This search mode can be restricted to the user's own bookmarks, all del.icio.us bookmarks or, since del.icio.us is part of Yahoo.com, to a web search. In case of the del.icio.us wide scope the results are divided into results from the user's bookmarks and all users' bookmarks. The returned results are ordered by posting time decreasing. A special feature is the search by media type, bookmarks of URLs that end

with specific extensions (e.g. doc, pdf, mp3, etc) are automatically tagged with "system:filetype:ext" where "ext" is the extension of the URLs filename. So it is possible to search for PDFs with system:filetype:pdf.

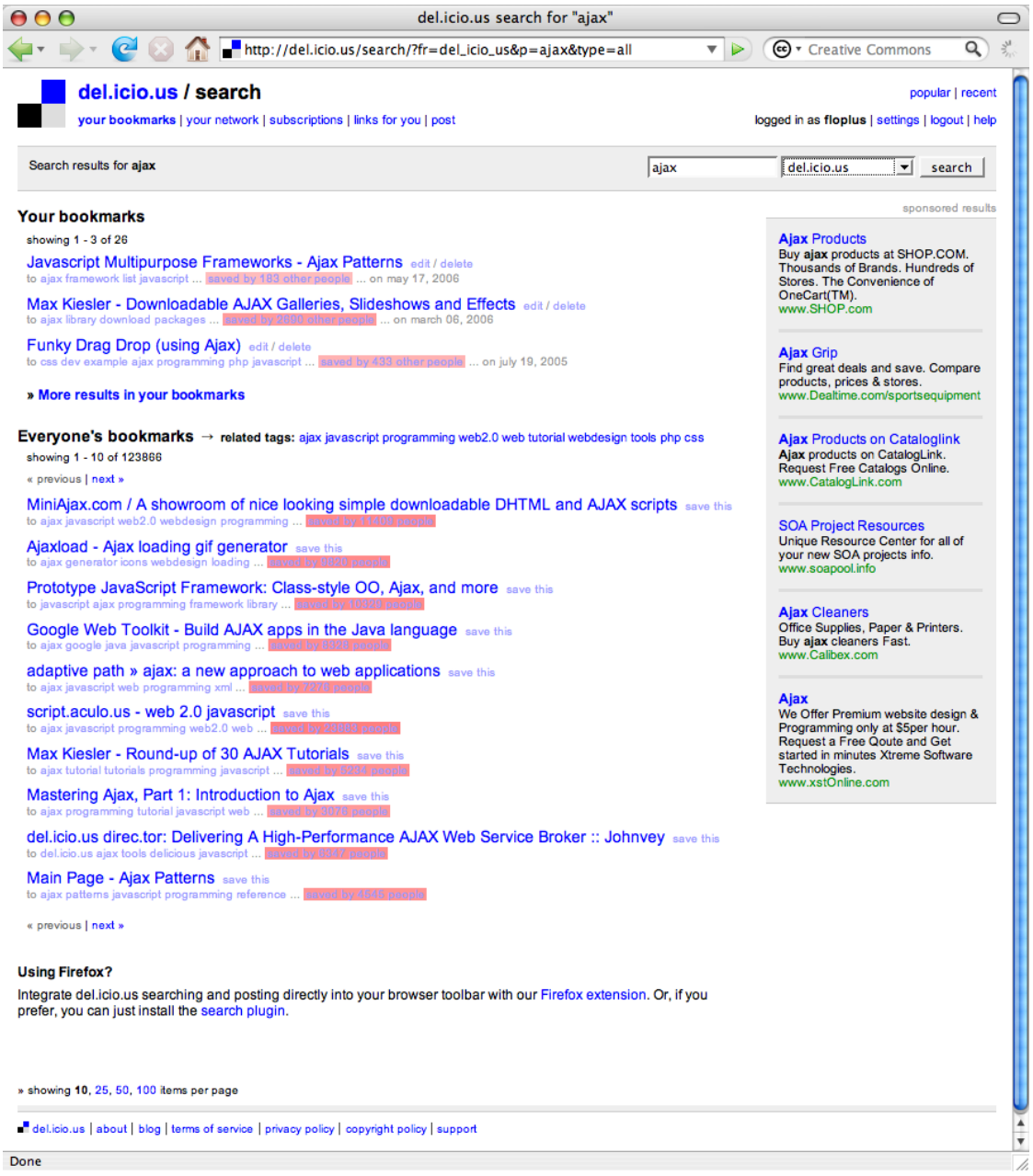

Illustration 6: Search result for the term 'ajax' on del.icio.us

7.2.3 Data Storage

A further requirement for the Social Bookmark extension results from the translation of the data storage to the client side. The del.icio.us service is responsible for the storage of the user's data (i.e. login data, bookmark data, the user's network

data) and is therefore able to provide search functionality both in the user's bookmarks and in all collected bookmarks and to connect the data sets of different users. If looked at the data attached to a URL only the for:XXX tags (the tags that hint a bookmark to another del.icio.us user) are used to support social networking, this means no special data must be collected to support basic hinting bookmarks to other users.

One bookmark exists of a field for each:

- URL
- Description of the bookmark
  Commonly the title of the page as a free form text
- A note for the bookmark
  Additional more detailed description of the page or anything the user wants to note to this page, free form text
- Tags
  This is a list of words separated by spaces, each word is a tag.
- Private
  A boolean field if the bookmark should be publicly visible on the del.icio.us website (false, not private) or not (true, private)
- First posted
  A date field that contains the date and time when the user has bookmarked this website for the first time

7.2.4 API

The del.icio.us API (del.icio.us, 2007c) works, as said, by making authenticated HTTP requests over SSL. The authentication mechanism is HTTP Basic authentication (Wikipedia, 2007d) and the authorization data is the user's login/password combination. The server response to a successfully authenticated request is a XML document. The methods available over HTTP requests are 'update', 'tags', 'posts' and 'bundles':

- update, retrieves the last time the user has posted an item
- tags, retrieves the users tags, or let a tag be renamed
- posts, retrieves the users posts (bookmarks), optionally filtered by tag, date,

URL; also adding and deleting of bookmarks is possible; it is not possible though to search for all users' bookmarks.

- bundles, are named collections of tags, which can be retrieved, created and deleted

A limitation of the del.icio.us API is, that it provides no search in all users bookmarks.

7.2.5 Summary

Del.icio.us has a relatively simple approach to bookmark management. Creation of a bookmark is easy: One click to begin the bookmarking process, the stating of the user's tags and a second click to finish the bookmarking process. The tagging process is supported by recommendations of tags, additional data can be provided to describe and find a bookmark again and a bookmark can be declared private.

Searching is done with similar ease: With the statement of one or more tags the user bookmarks are filtered by that tag and additionally the public tags from other users are also searched by the given tags.

The data that is stored by del.icio.us is also the basis for the data that the plug-in has to save.

The del.icio.us API provides functionality for creating, editing and deleting bookmarks and can therefore be used to implement the *Social Bookmark* concept.

7.3 Requirement Specification

The theory and analysis from the previous chapters are used to define the requirements of the *Social Bookmark* Firefox extension in the following section.

7.3.1 Human Requirements

The target user group are people who use the del.icio.us service and are concerned about their privacy, but want to benefit from the advantages of social bookmarking.

7.3.2 Functional Requirements

The functional requirements describe the functions of the *Social Bookmark* extension namely creating a bookmarking, searching for bookmarks. Before these actions are described the user interface concept and the data storage is specified first.

7.3.2.1 User Interface Concept

Bookmarking must not interfere with the Web browsing of the user more than necessary.

The user interfaces for the functions bookmarking, searching and synchronization are embedded (inline) in the browser window (with the exception of the preferences), this reduces the visible space of the webpage during an action, but both Web site and bookmarking interface are visible at the time. Such an arrangement facilitates the user's task as they stay in their context and do not have to switch between two windows (one with the page and one with the user interface for e.g. bookmarking).

To make the functions of *Social Bookmark* accessible to the user three most common ways are used: An entry in the 'Tools' menu, toolbar buttons, and global keyboard shortcuts. The first two options are recommendations of “Extension Etiquette” at the Mozilla Developer Center (Mozilla.org, 2007b), global keyboard shortcuts, called “hotkeys” (Mozilla.org, 2007c), can help speeding the access of a function but are harder to learn and can not be relied on in an environment, where possibly many extensions exist side by side and define the same keyboard shortcuts.

7.3.2.2 Data Storage

The bookmark data is stored in a database that understands a subset of SQL92 (see SQL92, 1992 and SQLite, 2007b). Bookmarks and tags are stored in separate tables in the form of a relational database (see Illustration 7). A row in the bookmarks table (BOOKMARKS) holds only the data that is unique for one bookmark, the tags are stored in a different table (TAGS); bookmarks and tags are related to each other by a intermediate table (BOOKMARKSTAGS) that stores the relation between a bookmark and its tags. This relationship is also needed for bookmark retrieval, as the bookmarks that use a tag are easily found.

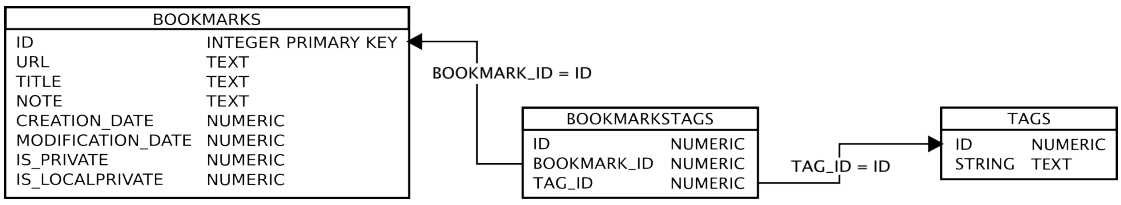

Illustration 7: Database schema of SocialBookmark

The data that is stored for a bookmark is derived from the data that must be provided on del.icio.us.

A bookmark therefore consist of:

- Id
  An Id that identifies the bookmark in the database, this is unique for all bookmarks.

- URL
  The URL of the page that this bookmark represents, there can only be 1 bookmark of a page.

- Title

A Title for the bookmark, this can be the Title of the page but this is not necessary.

- Description

The description is a Text field where the user can write additional information regarding the page.

- Creation date

The date when the bookmark is created; it never changes as long as the bookmark exists.

- Modification date

Additionally a bookmark has a modification date which is set, when an existing bookmark is changed.

The list of tags this bookmark is described with is NOT part of the bookmark table but mapped in a relation table. The data for a tag is:

- Id
  The id of the tag in the tags table; this is unique.

- String
  The string of the tag.

With the data and its storage now defined the activities of the user and the responses of the extension can be specified.

7.3.2.3 Bookmarking

The user interface for bookmarking is an inline panel, which is responsible to gather the needed information and is helping the user to provide the data: When bookmarking a page the user must be supported by automatically filling out needed data that can be collected by the software. For each of the bookmarks data fields, a

text input field must be present to allow user input. The date fields 'creation date' and 'modification date' are set automatically without an option to change them, because there is no need to. The tagging input is supported by recommending completions to a tag the user has begun to type. The recommendations are taken from previously entered tags as they are available. Other methods of tag recommendations (like popular tags) are discarded, because this would slow down the tagging process by network requests to del.icio.us.

7.3.2.4 Searching

The user interface for searching is also an inline panel; this allows for quick access to the user bookmarks without leaving the browsing window. The bookmarks are presented as a list of bookmarks with the newest bookmark at the top. It is assumed that newer bookmarks are generally more important as the context in which they were created (or modified) is more vivid. Retrieval of bookmarks is done by specifying tags which filters the list for bookmarks with the given tags. The same recommendation method for tags as for bookmarking is used, recommendations for other peoples' tags would not make any sense as they are not present in the user bookmarks/tags. When a user types a combination of tags that yield no result in their own collection of bookmarks they are offered a link to search on the del.icio.us website with the given tags. This is a confession to the fact that search in all users bookmarks can not be performed via the del.icio.us API (see 7.2.4 API).

7.3.2.5 Synchronization

The user must have an option to synchronize their bookmarks between the *Social Bookmark* instance and the del.icio.us service. To follow the user interface concept for bookmarking and searching this functionality also resides in an inline panel. Synchronization is initialized manually by the user, because there is no sensible way for the software to find out when this should happen. As synchronization can be a lengthy process the user must be informed about its progress.

7.3.2.6 Authentication

As the use of del.icio.us requires authentication a user interface must be present to get the user data (username and password). As this data normally does not change it is stored in the user's browser preferences and presented to the user when authentication at the del.icio.us service fails.

7.3.3 System Model

Following the functional specifications of the browser extension in its parts the

whole system model is presented in Illustration 8. The illustration gives an overview of the extension embedded into the Firefox browser.

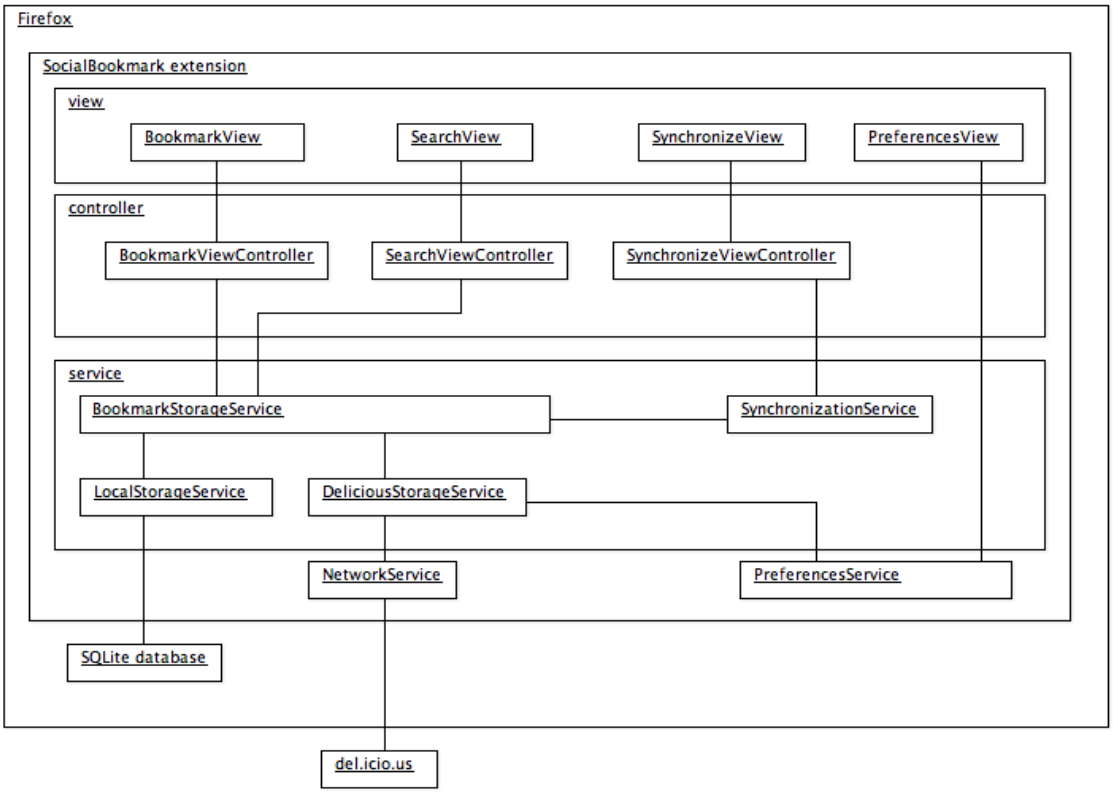

Illustration 8: Extension system overview

The extension architecture follows a Model-View-Controller (MVC) (Buschmann et al., 1996) pattern. The idea of this pattern is a separation between the data and its logic (model) from its display (view) and the control instance (controller) that receives user input and updates the model, which in turn updates the view.

The social bookmark extension can be divided in view logic, controller logic and service logic (this represents the model in the MVC). Additional services are used to abstract the Firefox JavaScript API for network access and preference storage.

Each function has a view (bookmarking, searching and synchronizing) and each view has a controller associated. The controller is responsible for managing the data that is displayed and collecting data from the view to forward to the services. A service is responsible for a collection of functions. The BookmarkStorageService is responsible for retrieval and storage of bookmarks, it distributes the bookmark data from the user interface to the storage services (LocalStorageService and DeliciousStorageService) and collects bookmark data from them. The SynchronizationService determines what kind of synchronization must take place (local to remote or remote to local) and carries it out by utilizing the

BookmarkStorageService. The LocalStorageService and the DeliciousStorageService are responsible for storing and retrieval of the bookmarks in the SQLite database and the del.icio.us Web site respectively. The DeliciousStorageService makes additional use of the NetworkService and the PreferencesService. The NetworkService initiates and handles the HTTP requests and responses from the del.icio.us API calls. The authentication information is obtained from the PreferencesService.

7.4 Implementation

This chapter shows the implementation of the Firefox extension based on 7.3 Requirement Specification. First the technical framework is explained and then screenshots with explanations describe the implemented functions.

7.4.1 Technical framework

The browser extension is developed for the Firefox browser starting from Version 2.0. This constraint is necessary because the data storage is based on a relational database model implemented in SQLite which is available from the Version 2 branch of the browser only. The tested operating system platforms are Microsoft Windows and Apple Mac OS X, though the extension should work on other platforms, too.

Development in the Firefox browser is mainly done in XUL and JavaScript[24]. XUL is an "XML-based user interface language" (Bojanic, 2007). XUL is used to define the user interface, the Firefox application takes this definition and displays the user interface. This approach has the advantage that the user interface is independent of the operating system platform it runs on, this is in the responsibility of the browser. Additionally the presentation can be accommodated with CSS[25] and the user interface can be changed by JavaScript via the DOM[26] representation of the XUL document. The user interface classes in Illustration 8 are built using XUL.

JavaScript is used to implement the extension logic and the service classes for the Social Bookmark extension (see Illustration 8). The SQLite database was accessed through a JavaScript interface provided by the Firefox browser.

The icons used for buttons are based on the *"Sweetie Icon Set"* (North, 2006).

7.4.2 Implemented Functions

This chapter explains the functions of the implemented *Social Bookmark* browser extension. Basic knowledge how an extension is installed in Firefox is assumed. The screenshots illustrating the explanations are taken in Firefox 2.0.0.4 on the Apple

24 JavaScript Version 1.7 (Mozilla.org, 2007d) is based on the ECMA standard 262 (ECMA, 1999).
25 For Cascading Style Sheets (CSS) specification see (W3C, 2006).
26 For Document Object Model (DOM) see (W3C, 2005).

Macintosh platform with the *Social Bookmark* extension version 0.1.8 released on the 8th of July, 2007 at the URL: http://fatalelogik.org/socialbookmark/.

7.4.2.1 Accessing the Extension Functions

After a successful installation in the 'Bookmarks' menu a new menu item 'Social Bookmark' is available, which has a sub-menu to access the main functions of the extension: 'Bookmarking...', 'Bookmark List...'[27], 'Synchronize...' and 'Preferences...' (see Illustration 9). The dots after the texts hint that a dialog follows after activating the menu item.

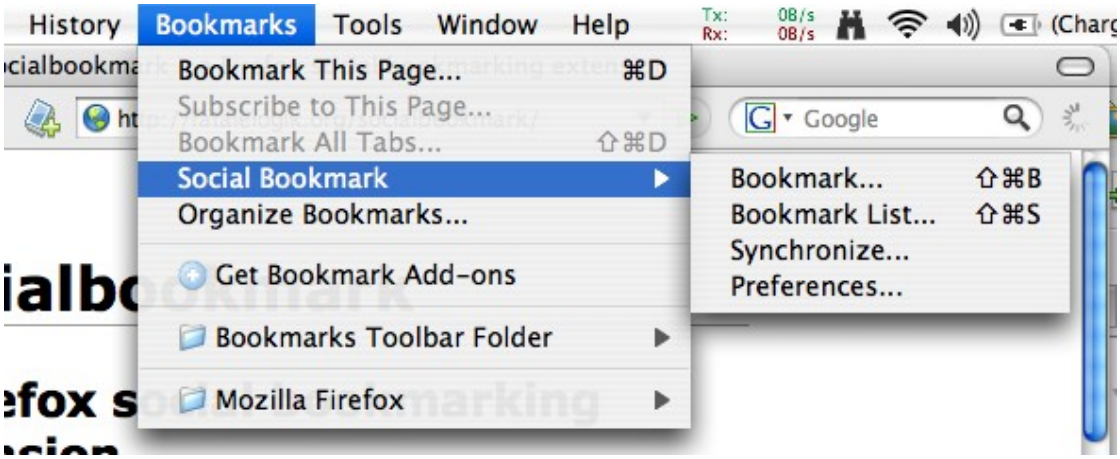

Illustration 9: Firefox 'Bookmarks' menu

The screenshot shows also the key combinations for the two main functions:

'Bookmark...' with APPLE + SHIFT + B (CTRL + SHIFT + B on Windows) to create or edit a bookmark for a page (see 7.4.2.3 Bookmarking), and

'Bookmark List...' with APPLE + SHIFT + S (CTRL + SHIFT + S on Windows) to show the list of bookmarks with search capability (see 7.4.2.4 Searching for Bookmarks).

A third possibility to access the extension functions is by toolbar buttons. These must be configured by hand via the 'Customize...' command in the context menu of the browser toolbar. In the customize dialog three new buttons are available.

Illustration 10 shows the buttons in the browser toolbar between the 'Home' button and the location field. These are from left to right: 'Synchronize', 'Bookmark List' and 'Bookmark'.

Illustration 10: Toolbar buttons

27 One obvious difference in the wording must be highlighted: The search functionality is available under the term 'bookmark list', because the dialog that follows holds a list of the user bookmarks, which can be searched by filtering with tags. A 'Search bookmarks' menu item would have been misleading because it would have disguised the fact that the user can see all their bookmarks with this command.

7.4.2.2 Synchronization with del.icio.us

After installation there are no bookmarks available, but it is possible to import bookmarks from the del.icio.us Web service with the synchronize functionality of Social Bookmark. Before this the account information for del.icio.us must be stated. This can be done via the extensions preferences dialog (see Illustration 11) and is saved in the user profile so it must be stated only once. An important note is that it is necessary to activate the 'private saving' option on the settings page of del.icio.us to use the private functionality of the website, unfortunately this can not be done via the API.

Illustration 11: Social Bookmark preferences

After giving the correct login information the extension can import the bookmarks from del.icio.us. This is initiated by calling the 'Synchronize...' menu item. A 'Synchronization' dialog appears at the bottom of the browser window, with two options, 'Synchronize' and 'Update'. 'Synchronize' tries to find out the last user intervention onto local as well as del.icio.us-stored bookmarks and then saves the local bookmarks to del.icio.us or reads them from del.icio.us to the local database respectively. 'Update' reads all bookmarks on del.icio.us and inserts (or replaces) the bookmarks to the local database. This can be used for initial synchronization and for

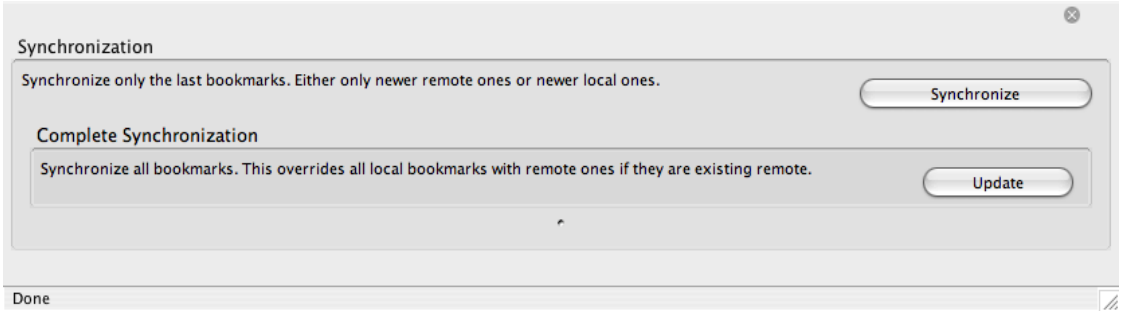

Illustration 12: Synchronize dialog

cases where synchronization does not work as expected.

Synchronization can be initiated by the user whenever desired. To keep the local and remote storage consistent, it makes sense to initiate a synchronization when the del.icio.us Web site was used to create a bookmark, or when bookmarks were created while not being online.

During the synchronization process the update progress is shown, which can also be canceled (see Illustration 13).

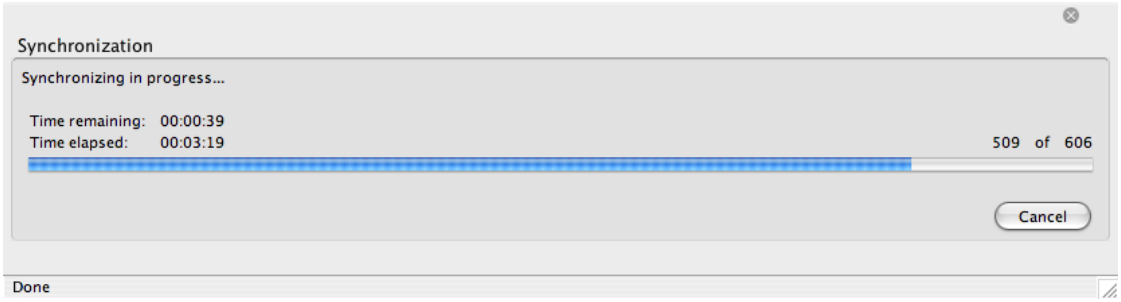

Illustration 13: Synchronize in progress

7.4.2.3 Bookmarking

shows how a bookmark is created. The user has selected a part of the website ('This is a Firefox browser extension....') and activated the 'Bookmark' function. The bookmarking dialog appears at the bottom of the browser window with prefilled input. The 'URL' is taken from the current browser window, the 'Title' it taken from the current browser windows title, and the 'Note' is the user selected text. The further options are the privacy settings and the 'Tags'. The user can set the privacy settings for this bookmark with the check-boxes on the right of the 'Notes' text field. They can declare the bookmark as a 'Private Bookmark' afterwards it is saved to del.icio.us (and the local database), but with the private marker set, or as 'Local private Bookmark', which prevents the bookmark from being sent to del.icio.us.

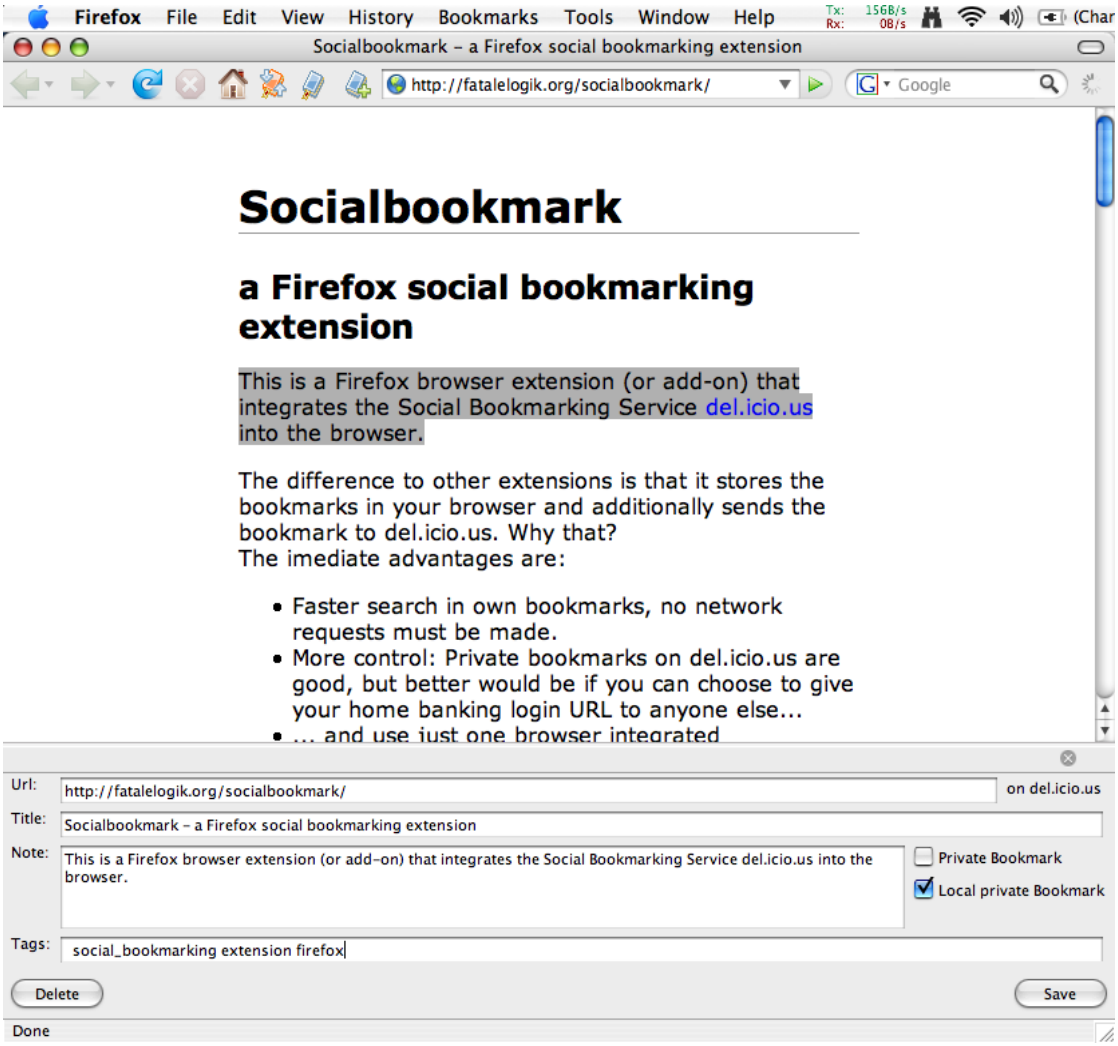

Illustration 14: Creating a bookmark

The 'Tags' are given by the user as simple space separated words. When they begin to type they receive hints with possible completions of the tag which should ease the choice of a tag and make the user's tags more consistent (see Illustration 15).

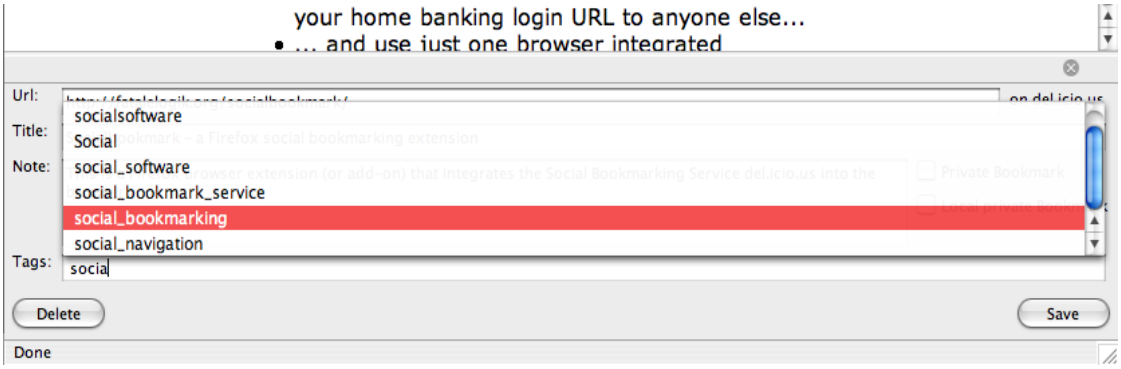

Illustration 15: Bookmarking tag completion

The bookmark is created by clicking the 'Save' button, if the bookmark should not be created the user can click on the 'X' button on the top right of the panel or by clicking the 'Delete' button. As this dialog is also used when editing an existing bookmark the 'Delete' button then deletes the bookmark from the bookmark collection (local and remote).

One further option is visible: the 'on del.icio.us' text on the right of the 'URL' field. Clicking on this text will open a page on del.icio.us with information how this URL is bookmarked on del.icio.us by all users[28] in a new browser tab.

7.4.2.4 Searching for Bookmarks

The retrieval of bookmarks is initiated by calling the 'Bookmark List' function. A dialog at the bottom of the browser window appears that holds a list of the users local bookmarks (see Illustration 16). Initially no bookmarks are filtered but for performance reasons only the 40 most current bookmarks are shown.

The bookmarks are shown in a vertical list, with one bookmark in each row. A bookmark is displayed with all information 'URL', 'Title', 'Tags', 'Note', privacy status and creation/modification date. The texts have different sizes that represent the assumed importance for the user. The 'URL' as the most abstract information smallest and the 'Title' as short, and therefore fast comprehensible, the biggest. The 'Note' field is only visible in case a note exists for a bookmark not to clatter the user interface.

At the right side of each bookmark is a button to edit the bookmark. A click on it calls the 'Bookmark' dialog for the bookmark and closes the bookmark list, to free screen space. Illustration 16 also shows the difference of a local private and private/public bookmark. A local private bookmark[29] has only an edit button, else[30] additionally the bookmarks info page on del.icio.us can be called clicking the 'more info' label corresponding to the 'on del.icio.us' function when bookmarking (see 7.4.2.3 Bookmarking).

28 This feature is not derived from the requirements specification but was implemented because it was simple and a possible step of integrating the social navigation aspect of social bookmarking.
29 The first in Illustration 15.
30 The second and third bookmark in Illustration 15.

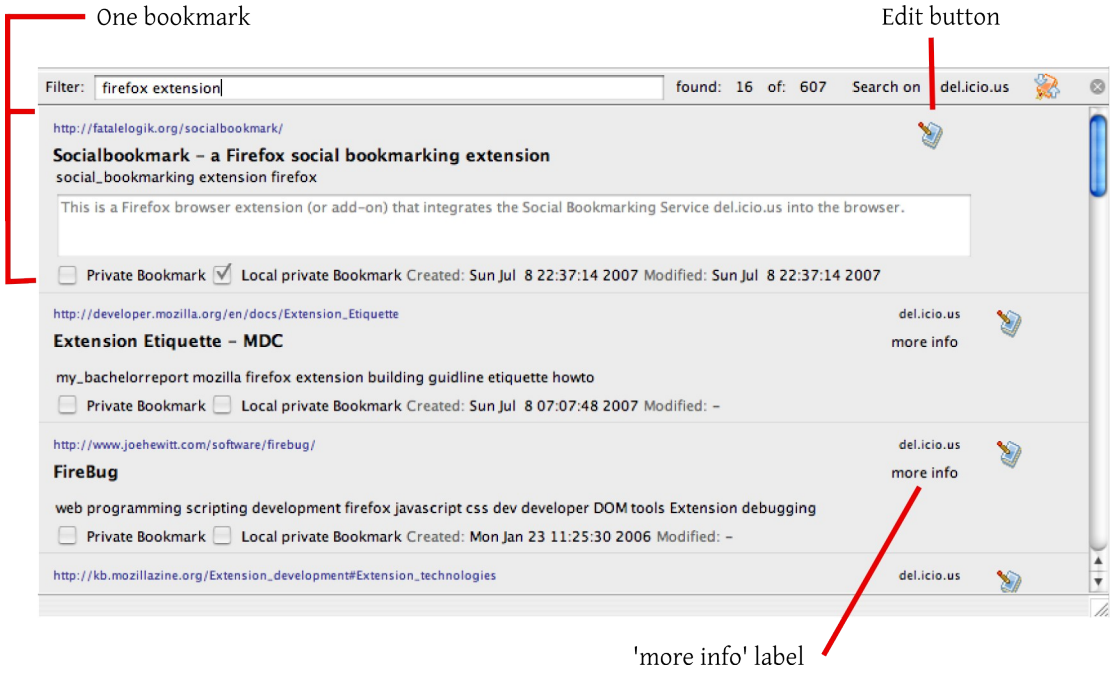

Illustration 16: Bookmark list

A bookmarks URL is called in the active browser tab by double clicking on the bookmark in the list; the bookmark list is then automatically hidden. When holding down the APPLE (on Windows the CTRL key) while double clicking the URL is opened in a new browser tab and the bookmark list stays open. This can be useful when the user wants to view several URLs from a search result.

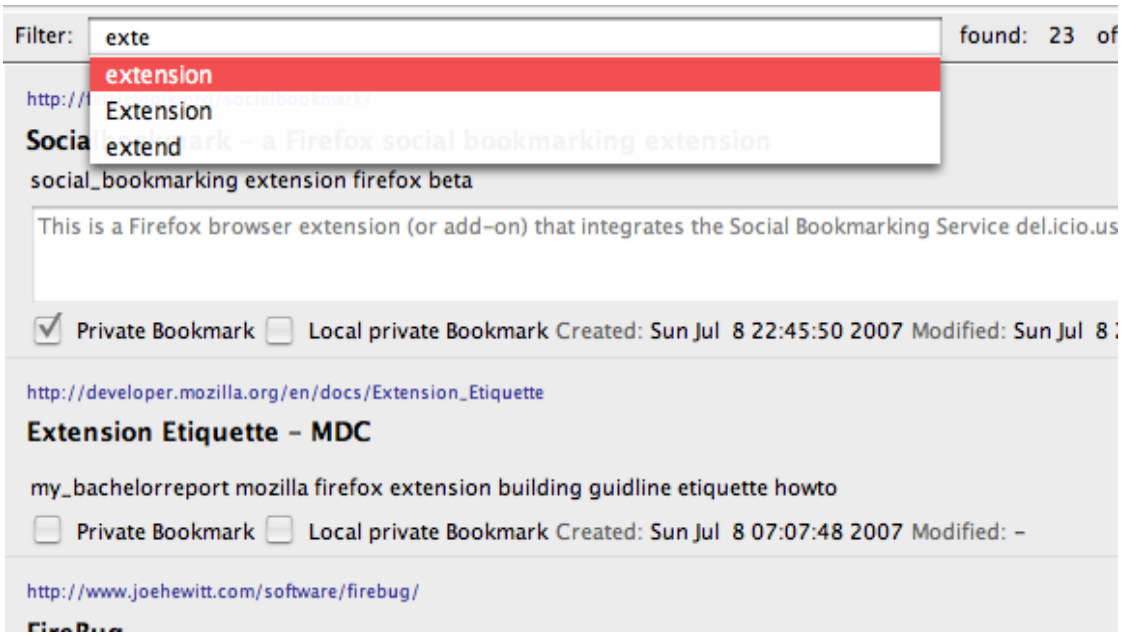

Illustration 17: Bookmark list tag completion

At the top of the dialog a 'Filter' input field allows the user to narrow the displayed bookmarks by stating tags, hinting of tags is the same as for creating bookmarks (see Illustration 17). To the right of the 'Filter' field the number of tags found to the given filter and the total number of the users bookmarks is shown. The list is filtered by the given tags as they are typed, even non complete tags focus the search result (see Illustration 17). If an empty result is the outcome of a filter, because the user does not have a bookmark with this combination of tags, they are offered to search the bookmarks on the del.icio.us website with the given tags (see Illustration 18).

Illustration 18: No search result in local bookmarks

The same can be achieved by clicking the 'Search on del.icio.us' label to the right of the filter field (see Illustration 19).

As a convenience the synchronize function can also be called in the bookmark list by clicking on the synchronize button with the red bookmark icon. The round 'X' button closes the bookmark list panel.

Illustration 19: Additional functions of the bookmark list

8 Conclusion and Outlook

This chapter will round up the findings of this work and hint where further work could be directed.

People are using their Web browser to navigate through the vast information space called the Internet. The directions come from other people, printed paper, search engines, etc., and when a person finds a place interesting enough that they wish to visit it again, they need a tool with which to find it the next time, a shortcut. Such a tool is in principle a bookmark to the place of interest (i.e. a page in the World Wide Web).

Eventually the person has a collection of bookmarks which must be made navigable. One way is to give each bookmark one word (or more) that comes to mind and represents a connection between the person and the page (tagging). Later the person can find the bookmark (and the page) by stating their interest, and using this connection.

As people are social beings, they want to communicate and exchange information about their interests. A person can use a social bookmarking website to manage their bookmarks, share them with others, in turn using the bookmarks of others to enrich their own collection.

This collection of bookmarks is of value to the person (e.g. saving time) and represents their personal interest. However although people want to communicate some things with the public they wish to do it with discrimination. People want to control what they share with whom and how. To prevent the splitting of their collection of bookmarks in different places, public and private, the proposed system *Social Bookmark* comes into operation.

With *Social Bookmark* a person stores their bookmark collection in their Web browsing tool, the browser, retaining full control over it. From here, they can manage their bookmarks and decide which to share and which to keep private.

To determine the feasibility of the concept, a prototype implementation has been produced. Though the prototype has been implemented with a reduced feature set (only one social bookmarking service,) the main feature of local bookmark storage (for control) with concurrent online storage (for use of social bookmarking features) has been realized.

With this, the goal of this bachelor report has been achieved.

Further research could go in several directions. One would be a survey about the

privacy preferences of social bookmarking websites users. This would clarify the users' demands for a system like this, and give hints how to refine the concept. How can the user be supported in the strive to maintain privacy? Is it feasible to develop algorithms making it harder to re-identify people using social bookmarking services? An obvious direction for study would be a usability evaluation of the extension presented here. What features are missing? Should service specific features be supported? This could also include a usage evaluation to find out how the browser extension influences the bookmarking behavior of users.

The Internet, as compared to the human civilization in terms of age, has just been born, yet makes a vast amount of data available at our fingertips. The dimensions of information and the way we generate and consume information are new to us. We need to develop strategies and technologies to deal with this new situation. This work is an attempt to ease our personal orientation on the Internet, in a way that makes us feel safe.

9 Bibliography

Abrams, D. (1997). *Human Factors of Personal Information Spaces*. Master of Science Thesis. University of Toronto.

Abrams, D., Baecker, R., Chignell, M.H. (1998). Information Archiving with Bookmarks: Personal Web Space Construction and Organization: In: *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems*, New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., pp. 41 - 48

Bellotti, V. & Sellen, A. (1993). Design for Privacy in Ubiquitous Computing Environments: In: *ECSCW '93: Proceedings of the Third European Conference on Computer Supported Cooperative Work (ECSCW'93)*, Norwell, MA, USA: Kluwer Academic Publishers, pp. 77 - 92

Bielenberg, K. & Zacher, M. (2005). *Groups in Social Software: Utilizing Tagging to Integrate Individual Contexts for Social Navigation*. Master of Science in Digital Media. Universität Bremen.

Boardman, R. & Sasse, A.A. (2004). "Stuff goes into the computer and doesn't come out": a cross-tool study of personal information management : In: *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, New York, NY, USA: ACM Press, pp. 583 - 590

Bojanic, P. (2007). *The Joy of XUL - MDC*. [Online]. Available from: http://developer.mozilla.org/en/docs/The_Joy_of_XUL [Visited: 08/07/2007].

Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., Stal, M. (1996). *A System of Patterns*. Chichester, England: John Wiley & Sons

del.icio.us (2007a). *del.icio.us*. [Online]. Available from: http://del.icio.us [Visited: 08/06/2007].

del.icio.us (2007b). *del.icio.us/about*. [Online]. Available from: http://del.icio.us/about/ [Visited: 07/06/2007].

del.icio.us (2007c). *del.icio.us/help/api*. [Online]. Available from: http://del.icio.us/help/api/ [Visited: 07/14/2007].

Dieberger, A. (2003). *Social Connotations of Space in the Design for Virtual Communities and Social Navigation*

Donath, A. (2007). *Pornos für alle: Sicherheitspanne bei Fotodienst flickR.com*. [Online]. Available from: http://golem.de/0702/50605.html [Visited: 07/06/2007].

ECMA (1999). *ECMAScript Language Specification*. [Online]. Available from: http://www.ecma-international.org/publications/standards/Ecma-262.htm [Visited: 07/14/2007].

El-Beltagy, S., DeRoure, D. & Hall, W. (1999). A Multiagent system for Navigation Assistance and Information Finding: In: *THe Fourth International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology*, London, UK: Commonwealth Institute, pp. 281 - 295

Encylopedia Britannica (2007). *art collection*. [Online]. Available from: http://www.britannica.com/ebc/article-9355861/art-collection [Visited: 07/09/2007].

Foner, L.N. (1999). *Political Artifacts and Personal Privacy: The Yenta Multi-Agent Distributed Matchmaking System*. Doctor of Philosophy. Massachusetts Institute of Technology.

Forsberg M., Höök, K. & Svensson, M. (1998). *Design Principles for Social Navigation Tools.* [Online]. Available from: http://ui4all.ics.forth.gr/UI4ALL-98/proceedings.html [Visited: 06/07/2007].

Frankowski, D., Cosley, D., Sen, S., Terveen, L., Riedl, J. (2006). You Are What You Say: Privacy Risks of Public Mentions: In: *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, New York, NY, USA: ACM Press, pp. 565 - 572

Golder, S. & Huberman, B.A. (2006). The Structure of Collaborative Tagging Systems. *Journal of Information Science*. 32(2) pp. 198 - 208

Halvey, M & Keane M.T. (2007). An Assessment of Tag Presentation Techniques: In: *WWW '07: Proceedings of the 16th international conference on World Wide Web*, New York, NY, USA: ACM Press, pp. 1313 - 1314

Hammond, T., Hannay, T., Lund, B. & Scott, J. (2005). *Social Bookmarking Tools (I).* [Online]. Available from: http://www.dlib.org//dlib/april05/hammond/04hammond.html#n1 [Visited: 06/07/2007].

Hassan-Montero, Y. & V. Herrero-Solana, V. (2006). Improving Tag-Clouds as Visual Information Retrieval Interfaces. To appear in: *International Conference on Multidisciplinary Information Sciences and Technologies, InSci2006*. Merida, Spain: Open Institute of Knowledge. Available from: http://www.nosolousabilidad.com/hassan/improving_tagclouds.pdf

Hein, R. (2007). *TheCounter.com: The Full-Featured Web Counter with Graphic Reports and Detailed Information.* [Online]. Available from: http://www.thecounter.com/stats/2007/June/browser.php [Visited: 06/30/2007].

Hipp, D.R. (2007). *SQLite home page*. [Online]. Available from: http://sqlite.org/ [Visited: 07/08/2007].

Hipp, D.R. (2007). *SQL Features That SQLite Does Not Implement*. [Online]. Available from: http://sqlite.org/omitted.html [Visited: 07/08/2007].

Iskold, A. (2006). *The Social Bookmarking Faceoff.* [Online]. Available from: http://www.readwriteweb.com/archives/social_bookmarking_faceoff.php [Visited: 07/05/2007].

Jäschke, R., Hotho, A., Schmitz, C. & Stumme, G. (2006). Wege zur Entdeckung von Communities in Folksonomies: In: *Proc. 18. Workshop Grundlagen von Datenbanken*, Halle-Wittenberg: Martin-Luther-Universität, pp. 80 - 84

Kanawati, R. & Malek, M. (2002). A Multi-agent System for Collaborative Bookmarking. To appear in: *Proceedings of the Fourth International Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS-2002 at AAMAS*02)*. Bologna, Italy: Paolo Giorgini, Yves Lespérance, Gerd Wagner, Eric Yu (Eds). Available from: http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-59/

Kanawati, R. and Malek, M. (2000). *Informing the Design of Shared Bookmark Systems*. [Online]. Available from: http://citeseer.ist.psu.edu/kanawati00informing.html [Visited: 09/07/2006].

Keller, R.M., Wolfe, S.R., Chen, J.R., Ritz, J.L. & Mathe, N. (1997). A Bookmarking Service for Organizing and Sharing URLs: In: *Selected papers from the sixth international conference on World Wide Web*, Essex, UK: Elsevier Science Publishers Ltd., pp. 1103 - 1114

Kohavi, R. (2001). Mining E-Commerce Data: The Good, the Bad, and the Ugly: In: *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM Press, pp. 8 - 13

LookSmart (2007). *LookSmart's Furl - Your Personal Web*. [Online]. Available from: http://www.furl.net/user/login?logout=true [Visited: 06/07/2007].

Martin Jr., D.M., Smith, R.M., Brittain, M., Fetch, I. & Wu, H. (2000). *The Privacy Practices of Web Browser Extensions*

Mozilla (2007). *Firefox - Rediscover the Web*. [Online]. Available from: http://www.mozilla.com/en-US/firefox/ [Visited: 07/08/2007].

Mozilla.org (2007a). *Extensions - MDC*. [Online]. Available from: http://developer.mozilla.org/en/docs/extension [Visited: 07/08/2007].

Mozilla.org (2007b). *Extension Etiquette - MDC*. [Online]. Available from: http://developer.mozilla.org/en/docs/Extension_Etiquette [Visited: 07/08/2007].

Mozilla.org (2007c). *Mozilla Keyboard Planning FAQ and Cross Reference*. [Online]. Available from: http://www.mozilla.org/access/keyboard/ [Visited: 07/14/2007].

Mozilla.org (2007). *New in JavaScript 1.7 - MDC*. [Online]. Available from: http://developer.mozilla.org/en/docs/New_in_JavaScript_1.7 [Visited: 07/14/2007].

NetApplications (2007). *http://marketshare.hitslink.com/report.aspx?qprid=0*. [Online]. Available from: http://marketshare.hitslink.com/report.aspx?qprid=0 [Visited: 06/30/2007].

Nielsen, J. (1990). *Hypertext and Hypermedia*. San Diego: Academic Press, Inc.

North, J. (2006). *Sweetie Icon Set*. [Online]. Available from: http://sweetie.sublink.ca/ [Visited: 07/14/2007].

Pirolli, P. & Card, S. (1995). Information foraging in information access environments: In: *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., pp. 51 - 58

Pitkow, J. (1998). *GVU's 10th WWW User Survey*. Atlanta, Georgia: GVU Center, College of Computing, Georgia Institute of Technology

Richter, A. and Koch, M. (2007). *Social Software - Status quo und Zukunft*. Munich, Germany: Fakultät für Informatik, Universität der Bundeswehr München

Rotenberg, M. (1993). Communications Privacy: Implications for Network Design. *Communications of the ACM*. 38 pp. 61 - 68

Rowley, J.E. (1992). *Organizing Knowledge*. Hants, England: Ashgate

Schmidt, J. (2007). Social Software: Facilitating Information-, Identity- and Relationship Management: In: *BlogTalks Reloaded. Social Software - Research \& Cases*, Norderstedt, Germany: Books on Demand, pp.

SQL92 (1992). *Database Language SQL*. [Online]. Available from: http://www.contrib.andrew.cmu.edu/~shadow/sql/sql1992.txt [Visited: 07/14/2007].

Sterling, B. (2005). *Order Out of Chaos* . [Online]. Available from: http://wired-vig.wired.com/wired/archive/13.04/view.html?pg=4 [Visited: 06/29/2007].

StumbleUpon (2007). *StumbleUpon :: Firefox Add-ons*. [Online]. Available from: https://addons.mozilla.org/en-US/firefox/addon/138 [Visited: 07/03/2007].

Upsdell, C.A. (2007). *Browser News: Statistics - find the browser stats you want to know*. [Online]. Available from: http://www.upsdell.com/BrowserNews/stat.htm [Visited: 06/30/2007].

W3C (2005). *Document Object Model (DOM)*. [Online]. Available from: http://www.w3.org/DOM/ [Visited: 07/14/2007].

W3C (2006). *Cascading Style Sheets, level 2 revision 1*. [Online]. Available from: http://www.w3.org/TR/CSS21/ [Visited: 07/14/2007].

W3Schools (2007). *Browser Statistics*. [Online]. Available from: http://www.w3schools.com/browsers/browsers_stats.asp [Visited: 06/30/2007].

Wikipedia Community (2007a). *AJAX*. [Online]. Available from: http://en.wikipedia.org/wiki/AJAX [Visited: 07/04/2007].

Wikipedia Community (2007b). *RSS*. [Online]. Available from: http://en.wikipedia.org/wiki/RSS [Visited: 07/04/2007].

Wikipedia Community (2007c). *Https*. [Online]. Available from: http://en.wikipedia.org/wiki/Https [Visited: 07/14/2007].

Wikipedia Community (2007d). *Basic authentication scheme*. [Online]. Available from: http://en.wikipedia.org/wiki/Basic_authentication_scheme [Visited: 07/14/2007].

10 Illustration Index

11 Appendix

11.1 Evaluation of social bookmarking services:

The three Web sites have been searched for a short self-description, a login has been created to get informed about:

- the personal data that must be provided to use the services (pseudonymity),
- if the service supports tagging,
- private bookmarking and if
- an API is announced by the operators of the Web site

After this a description was composed on the personal impression of the author.

11.1.1 stumbleupon.com

Self description (stumbleupon, 2007):	*StumbleUpon lets you "channelsurf" the best-reviewed sites on the web. It is a collaborative surfing tool for finding and sharing great sites. This helps you find interesting webpages you wouldn't think to search for."*
Description:	This service emphasizes on a collaborative experience of the Web. Their users "stumble upon" websites and rate them with a "thumb up" or "thumb down" if they like a website or not. Based on these recommendations, and categories a user chooses, a profile of interest is build and URL recommendations to the user is refined. The system allows the user to organize their URLs with tags and review it, but all URLs are public.
Pseudonymity:	username, email address, birthday and gender must be provided.
Tagging:	Yes
Private bookmarking:	No
API:	No

11.1.2 furl.net

Self description (furl.net, 2007): *An easier way to save and share your online discoveries. With Furl, you can:*

** Find Web sites that others have Furled*

** Save a copy of any web page with one click*

** Share anything you've Furled -- or keep it private*

Furl also lets you store and organize all of your bookmarks.

Description: The emphasis lays on building a personal bookmarks collection. Users can actively rate (0 to 5 stars), categorize, comment and add keywords (tags) to URLs, a copy of the website is saved at time of bookmark creation, which can be retrieved later. Export to various formats including academic bibliography formats like BibTeX or Endnote can be used to support research work.

Pseudonymity: Username, password and email address must be provided; name and first name are optional, the statements that the registering person is over 13 years old can probably be disregarded.

Tagging: Yes

Private bookmarking: Yes

API: No

11.1.3 del.icio.us

Self description (del.icio.us, 2007b): *del.icio.us is a collection of favorites - yours and everyone else's. You can use del.icio.us to:*

** Keep links to your favorite articles, blogs, music, reviews, recipes, and more, and access them from any computer on the web.*

** Share favorites with friends, family, coworkers, and the del.icio.us community.*

** Discover new things. Everything on del.icio.us is someone's favorite -- they've already done the work of finding it. So del.icio.us is full of bookmarks about technology, entertainment, useful information, and*

more. Explore and enjoy.

Description: del.icio.us focuses on the single users advantage of tagging bookmarks online, access them from anywhere, and searching for bookmarks. Additionally users can subscribe to other users bookmarks or recommend URLs to other users by tagging a bookmark for that user. Additionally bookmarks can have short notes.

Pseudonymity: Username, password and email address must be provided; the statement of the full name is optional.

Tagging: Yes

Private bookmarking: Yes

API: Yes

11.2 CD-ROM

Attached to this bachelor report is a CD-ROM with

- this text as a PDF file (Bachelor Report.pdf),
- the source code of the implemented extension (socialbookmark),
- a platform independent installer file (socialbookmark.xpi) and
- the current Mozilla Firefox software (version 2.0.0.4) for Microsoft Windows (Firefox Setup 2.0.0.4.exe) and Apple Mac OS X (Firefox 2.0.0.4.dmg).