Towards a Webpage-based Bibliographic Manager

Dinh-Trung Dang, Yee Fan Tan, and Min-Yen Kan

Department of Computer Science, School of Computing,
National University of Singapore, Singapore, 117590
`{dangdinh,tanyeefa,kanmy}@comp.nus.edu.sg`

Abstract. We present ForeCiteNote, an application that organizes personal digital collections of research articles. It is architected as a single HTML page with embedded Javascript that runs within a web browser. On top of standard annotation and tagging functionality, it also supports both online and offline usage patterns, including local storage of the paper collection.

1 Introduction

Scholarly information has greatly expanded in recent years, and while the exact balance of causes responsible for this are difficult to pinpoint, certainly open-access digital libraries and timely dissemination play a part [1]. While such free flow of information presents an unparalleled learning opportunity, it also presents challenges in managing these information sources, especially for beginning researchers.

To focus on the needs of researchers, we performed a focus group study at two academic institutions, involving 12 participants from the French National Institute for Research in Computer Science and Control (INRIA) and 7 from the National University of Singapore (NUS). Each participant completed ten questions on how they performed their own bibliographic management. The findings showed that around 60% store their papers both on- and off-line and have some difficulty organizing them. 43% percent reported using offline applications to take notes on research papers. Over 60% usually access their data when they are away, and do not view their own method as convenient nor systematic. 83% of those surveyed manually enter all the metadata (e.g., author, date, conference) when annotating papers. A good *bibliographic management application* (BMA) is thus needed for organizing documents: allowing researchers to collect, retrieve, annotate and organize citations and digital copies of scholarly articles.

2 Related Work

BMAs are not a new topic; in fact there are many commercial systems to cover such needs. A comprehensive survey of all solutions is beyond the scope of this paper. We focus on a few current major systems that represent the spectrum available. EndNote (`http://www.endnote.com/`) is the canonical example of the standalone BMA. It features comprehensive import and export functionality, organization of digital files, and word processing program integration. On the other hand, web-based applications such as CiteULike (`http://www.citeulike.org/`)[2] and BibSonomy

(`http://www.bibsonomy.org/`) [3] have central servers housing users' bibli-
ographies, thereby promoting sharing and collaborative annotation and tagging among
users. Connotea (`http://www.connotea.org/`) [4] also facilitates offsite web
use through browser extensions that can send annotations and references back to the
central server. Zotero (`http://www.zotero.org/`) takes this idea further by em-
bedding the whole BMA within a Firefox extension, eschewing a server based solution
entirely.

Each architecture has distinct advantages and compromises. A standalone applica-
tion offers the maximum flexibility but requires the user to run a separate application
and is prone to data loss (if the user's hard drive crashes). Server based BMAs make
the provider responsible for data integrity and immediately provide collaborative pos-
sibilities. However, this architecture cannot manage a user's local digital collection and
requires access to the web at all times. Browser extension based BMAs play well both
on- and off-line, but increase the memory footprint of the browser and also have the
problem of data loss. Also, as the system is tied to a specific web browser instance,
their library does not port with them if the user changes browser or computer.

From the above architectural discussions, four central issues emerge: 1) accessibil-
ity both on- and off-line, 2) portability across machines, 3) preventing data loss, and
4) local file management. ForeCiteNote aims to solve these issues, by basing its biblio-
graphic management on a different technology: the Javascript-enabled local web page.

3   ForeCiteNote: A TiddlyWiki-based Bibliographic Manager

ForeCiteNote is a client-side solution, with a supporting server component, that ad-
dresses the above issues. The server component provides data integrity by allowing
users to synchronize or backup their annotations and references. It also features online
access to the user's data and is capable of adding references and annotations, similar to
other server-based BMAss. The key difference is in ForeCiteNote's client-side compo-
nent, which is a modified version of a TiddlyWiki.

TiddlyWiki (`http://www.tiddlywiki.com/`) is a self-contained HTML page
which uses embedded Javascript to give the page the functionality to act as a wiki or a
blog. A user can add entries to keep notes or progress logs to the TiddlyWiki in entries
known as *tiddlers*. As it is a single, standard HTML file, it can be uploaded to a web
server so that it can be accessed anywhere with an internet connection, and downloaded
to other PCs, resolving Issue 1. When used off-line (locally on the user's computer),
the permissions afforded by the `file:` protocol allows new and modified tiddlers to
be saved by overwriting the TiddlyWiki itself on the user's disk. Since TiddlyWiki is
just a single, small (~400KB) web page and requires no installation, users can store
their own TiddlyWiki on a local USB drive and carry it around to view on any offline
computer equipped with web browser. This makes TiddlyWiki eminently portable and
thus resolves Issue 2. To resolve the remaining two issues, we had to modify the basic
TiddlyWiki, resulting in the final ForeCiteNote client, which we describe next.

To solve Issue 3, ForeCiteNote features a synchronization feature that allows the
client ForeCiteNote to send its annotations and entries to a ForeCiteNote server. The
synchronization engine first allows the upload any entries to the server and then allows
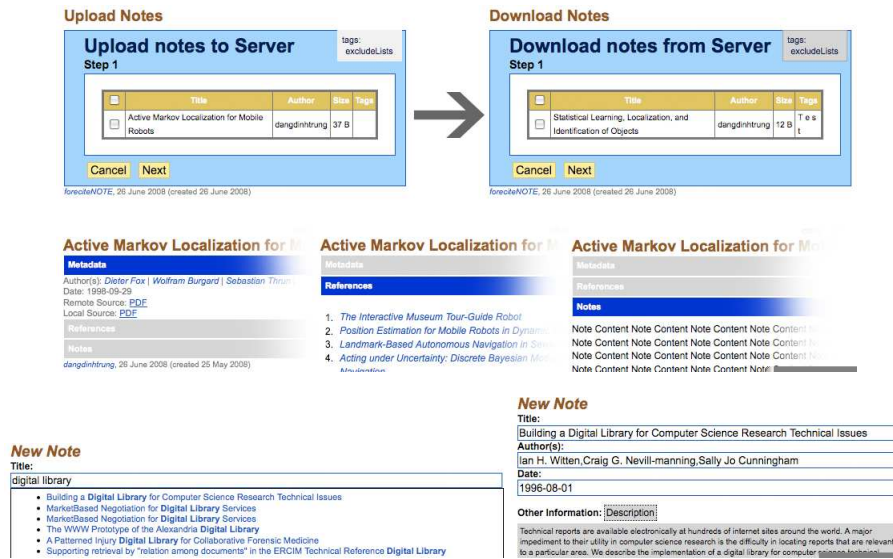
Fig. 1. Top row: Synchronization wizard. Middle row: New display format for tiddlers. Bottom row: Suggestion lists (left) and metadata autocompletion (right).

the user to download any missing entries (Figure 1, top row) illustrates this. In the drastic case where the user has lost their client-side web page, one obtains an empty client and synchronizes it to download all of their entries.

To solve Issue 4, the ForeCiteNote client also manages a directory where it stores local documents (named `paperlib`). As users may want to browse the local collection outside of the ForeCiteNote webpage application, care was taken to organize local documents in a useful and domain-independent way. According to [5], researchers identified the author, year and title as the three most prominent metadata that identify a scholarly document. ForeCiteNote structures its directory accordingly, in an iTunes-like directory structure: where the author and year are the first and second respective subdirectory level structures within the local document directory, and where the document itself is renamed after the title of the document.

For documents that the ForeCiteNote server knows the metadata and a location of an open access copy, the user can use the client to download and store a copy into the local document directory. The user can similarly ask the ForeCiteNote client to store a copy of other documents that the user has a copy, but which is unknown to the server. Sharing a document collection is as simple as sending the directory tree to a collaborator, which can be done by using archiving utilities and email. Also, since there are many desktop search applications, we intentionally left out local collection search functionality from the client.

We also customized ForeCiteNote to improve upon its usability for recording research notes. First, we modified ForeCiteNote's tiddler format, to better display and

alternate between displaying a work's metadata or its annotations in one place, as in Figure 1 (middle row).

Second, in our focus groups, metadata entry was identified as a severe bottleneck in annotation. Users clearly wanted references chunked into specific fields (e.g., author, title, year) but did not want to manually enter all of these fields. When the client has internet access, it can query the server to retrieve pertinent metadata to perform metadata autocompletion. When the user first creates a new entry for a paper, they start by typing in the document title. After three letters have been typed in, the client queries the server for any autocompletion matches based on papers known to the system. These are shown to the user, as in Figure 1 (bottom row, left). The user can then either ignore the suggested matching list or pick the document's title if shown. If a suggested known paper is chosen, the client retrieves the metadata from the server, including the abstract and any open-access location, and fills in the appropriate fields in the entry, as in Figure 1 (bottom row, right).

4   Conclusion

We have introduced ForeCiteNote, a webpage-based bibliographic manager. As an adaption of the revolutionary TiddlyWiki software, ForeCiteNote not only allows researchers to collect, annotate and organize citations and digital copies of scholarly articles and also a means to retrieve them, but also guarantees the accessibility both online and offline and portability across machines.

ForeCiteNote has undergone several rounds of informal user design testing and has incrementally been improved to its current state, reported here. Current work focuses on a long-term, longitudinal assessment of ForeCiteNote's usability. We expect to release the application for beta testing soon. Future work includes extending the ForeCiteNote server functionality to support the collaborative work of distributed groups.

References

1. Prosser, D.C.: Scholarly communication in the 21st century - the impact of new technologies and models. J. Serials Community **16** (2003) 163–167
2. Emamy, K., Cameron, R.: CiteULike: A researcher's social bookmarking service. Ariadne (2007)
3. Hotho, A., Jschke, R., Schmitz, C., Stumme, G.: Bibsonomy: A social bookmark and publication sharing system. In: Conceptual Structures Tool Interoperability Workshop at 14th ICCS Conf. (2006) 87–102
4. Anonymous: Join a social revolution. Nature **436** (2005) 1066
5. Day, M.Y., Tsai, T.H., Sung, C.L., Lee, C.W., Wu, S.H., Ong, C.S., Hsu, W.L.: A knowledge-based approach to citation extraction. In: IRI. (2005) 50–55