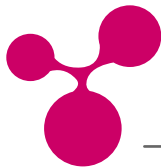


Technische Universität Dresden – Fakultät Informatik
Professur für Multimedialechnik, Privat-Dozentur für Angewandte Informatik

Prof. Dr.-Ing. Klaus Meißner
PD Dr.-Ing. habil. Martin Englien
(Hrsg.)



GENEME '09

GEMEINSCHAFTEN IN NEUEN MEDIEN

an der
Fakultät Informatik der Technischen Universität Dresden

mit Unterstützung der

3m5. Media GmbH, Dresden
GI-Regionalgruppe, Dresden
Communardo Software GmbH, Dresden
Kontext E GmbH, Dresden
Medienzentrum der TU Dresden
nubix Software-Design GmbH, Dresden
objectFab GmbH, Dresden
SALT Solutions GmbH, Dresden
Saxonia Systems AG, Dresden
T-Systems Multimedia Solutions GmbH

am 01. und 02. Oktober 2009 in Dresden

<http://www-mmt.inf.tu-dresden.de/geneme/>
geneme@mail-mmt.inf.tu-dresden.de

B.2 Social Feed Reader: Status Quo and Future Perspectives

*Martin Böhringer, Alrik Degenkolb, Andreas Schneider
Technische Universität Chemnitz, Fakultät für
Wirtschaftswissenschaften*

1 Introduction

The management of information is becoming an increasingly important topic, not only in society but also in the corporate landscape [Ch00]. Every day, one can read thousands of news stories on hundreds of websites – be they within the field of politics or about the release of a new software product. Now, with the rise of Web 2.0, that plethora of possibilities is becoming even bigger [Di04]. There are wikis, blogs and all sorts of technologies to ease the development and presentation of information systems. With many portable devices becoming web-aware, the Web is turning into a second layer of our daily lives, where we cannot only read and see but also write and show [Le05].

With this growing amount of information we have to find ways (and tools) to manage not only their presence but also their flow. On the one hand, we want to know more and be informed about important events, and on the other hand, we are not able to handle this huge stream of information by ourselves [HT01].

In this paper we will discuss feed readers as a possible solution for managing this information flow. We argue that common feed readers are mostly single-user tools. This is a surprising observation for a type of application that is mainly used to monitor social software sources. We evaluate XWiki Watch, an innovative but as yet unknown software product that uses a collaborative approach to feed management and could solve some of the information management issues at an enterprise level or at least within (work)groups. Afterwards we go further in discussing possible enhancements to this approach by using common social software techniques.

2 Background

2.1 Social Software and Web 2.0

Probably the most used buzzwords of the last few years are “Web 2.0” and respectively “social software”. While there is no exact definition of these terms, their core meaning is the development of Internet users from pure consumers to producers of content [HW05]. Being a buzzword also brings along the danger of being widely misused whenever some marketing expert sees an advantage in doing so [U108]. This leads to

fancy websites being labelled as “Web 2.0” although the real innovation is the concept behind the scenes. There are lots of new possibilities to communicate, not only within a group of people but also within a whole society. Blogs are used as some kind of publicly available diary [Na04] and, even on Ebay or YouTube, users communicate via comments [Pa08] to existing (or newly added) entries. This interrelation between users and producers as well as the fading border between them is what makes “Web 2.0” special.

2.2 Feed Formats

Web feeds – currently RSS and Atom – are methods of delivering information on demand. In contrast to email-based systems, the user does not have to provide any personal information. Instead, the client requests current items “on demand”, which also means that there is neither the necessity nor the possibility to unsubscribe from such a feed – if the client stops “asking”, the server won’t send any “response”. Another important concept is the standardised format in which the information is embedded. Websites are usually quite different from each other and can therefore not be processed in a generalised manner – different structures and layouts make it nearly impossible to extract any useful information in a universal way. Web feeds – also called web syndications – offer a generic format in which information can be presented and even transmitted. Of course, they don’t even have to be connected to any website at all; they could just as easily come from any other source like an enterprise resource planning (ERP) system or the like.

The web syndication format RSS was first introduced and used in 1999 in its version 0.9 by Netscape for use in the “My Netscape” portal site [Ne99]. At that time, it was called “RDF Site Summary” and can be imagined as a collection of XML schemes that define the structure of the document. This structure should be as generic and lightweight as possible to allow the embedding of a wide range of information. After Netscape abandoned RSS during a restructuring in 2001, several interested third parties took over – one of them being Dave Winer, who extended and redefined RSS for UserLand Software, which finally culminated in the release of RSS 2.0 in 2003 under the name “Really Simple Syndication”, which is now considered frozen: there should be no further changes in the specification that could break compatibility with older versions [Wi03].

Due to the aforementioned history of RSS, which has passed through many different hands and even involves drawbacks in favour of older versions (backward compatibility), there was the call for a new format that “does it right” straight from the beginning [Tr03]. Like RSS, Atom uses XML to embed the actual content into easily transferable code. In contrast to RSS, it uses the already existing possibilities better, like language differentiation (via “xml:lang”) and even encryption and signing on entry level (via the respective XML specifications). A complete and therefore much

more detailed comparison was performed by Sam Ruby, one of the driving forces behind the first steps towards a new format [Ru08]. The final specification, Atom 1.0, was published in 2004 under the standards body “Internet Engineering Task Force” as IETF RFC 428717.

3 Feed Aggregation

3.1 Why?

Usually, a user does not know which news in particular to read – otherwise it would hardly be “news”. To gain an overview about everything that is at least of some interest, it is necessary to aggregate all the feeds that the user wants. This means that all (new) items are combined into a single and possibly filtered view. Such filters could split the items according to keywords, patterns or simply by the website they were taken from.

In the end, this results in the user being able to watch and follow as many information sources as she likes. It avoids having to search and navigate through different websites and front-ends to deliver the desired content in a uniform and integrated manner.

3.2 Who?

The main interest in aggregation comes from individuals who want to stay up to date with recent developments – be they software engineers who would frequently visit several developer websites to be informed about new versions or hobby-computer users who just want to read recent “real world news” from several sources. Of course it doesn’t make much sense if a team of professionals (or individuals) reads the same sources but processes them all individually. The idea would be to aggregate them centrally and if possible even read them centrally. Surprisingly, this is not the case for common feed readers. If there are websites like YouTube that allow a video to be posted and users to comment on it, why shouldn’t it be possible to select interesting feeds collaboratively, aggregate them and share opinions about single items? Technically, it is already possible and will be discussed later in this paper.

3.3 How?

In simple terms, the aggregation is carried out by collecting recent items from selected feeds, filtering them according to the provided criteria and presenting them in a uniform way. This can either be performed by a fat client application or completely server side with a web front-end to manage and read the subscribed feeds. For a collaborative approach the last one is to be preferred since a server-based approach is necessary anyway and a web front-end allows easy access from different locations – if applied correctly even from mobile devices.

Since both RSS and Atom feeds are uniform information mappings embedded in XML it is possible to store them easily in a database. This provides two huge advantages: first, it is possible to aggregate the feeds regularly without having to query the feed providers whenever one of the users accesses the feed reader. Instead, the server carries it out automatically every specified amount of time (for example every five minutes) and stores the new entries in the database. Second, the databases can be optimized for fast data access and filtering. Therefore, the actual processing of user-defined requirements – like categorising or otherwise filtering – can be achieved on the fly via SQL statements.

4 An Example Application: XWiki Watch

Most feed reader applications do not offer any collaborative approaches to feed management. The well-known Google Reader comes with low-level social functionality like shared comments but it seems to be a long way from broad collaboration features such as those offered for example in Google Docs. However, we discovered XWiki Watch, a minor sub-project in the Open Source XWiki application with a social approach to feed management. The following paragraphs describe this piece of software in order to gain insights for the possible design of social feed readers.

4.1 XWiki Application Stack

XWiki is a web application written as a Java Web Application utilising an application server like Apache Tomcat and using Google Web Toolkit as the framework to build the core services and provide an interactive and extensible user interface.

Basically, XWiki is a wiki application that can be extended and recombined into several packages to suit different needs (see figure 1). An example of such a recombination is XWiki Enterprise, which already includes the necessary user front-end as well as business-related features like LDAP authentication – which is usually used in corporate networks – and PDF export – which is essential to make content available offline. Another example showing the extensibility is the product XWiki Watch. It enhances the wiki with a feed reader that can be managed by several users who can mark important items and comment on them.

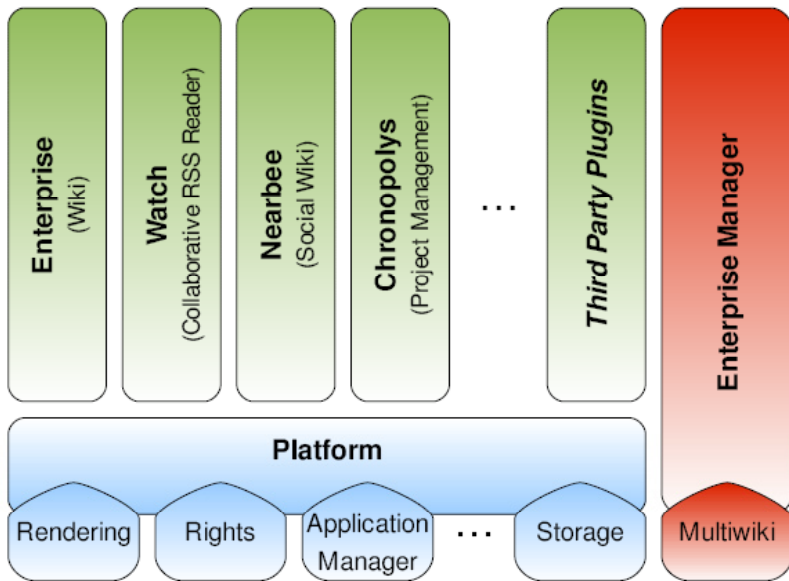


Figure 1: The XWiki architecture

The modular design also makes it possible for further layers to be laid out over the whole family of applications. The XWiki Enterprise Manager is such an example, providing the ability to manage agglomerations of wikis. That way, a complete wiki can be set up from a template, user accounts can be managed centrally and it is even possible to keep an eye on the statistics of the whole “farm”.

4.2 XWiki Watch

First, we start by looking at the most noticeable part, the user interface and the usability, before we dig deeper into the internals of this application and its source code. For this evaluation, we use the standalone version that is available from the XWiki website. This includes XWiki, the Watch application as well as the Jetty application server and an embedded and already preconfigured HSQL database. Therefore, it can be started with only a Java Runtime Environment present and will then be available from any web browser. If it is desired to be part of a production environment, it can also be integrated into an existing XWiki (Enterprise) setup and/or use any other application server (like Tomcat or Geronimo) and any other database-management system (like PostgreSQL or Oracle).

Welcome to XWiki Watch

Welcome to XWiki Watch ! XWiki Watch will allow you to track information about your company or the subject that you are interested in in Real Time. It will also allow you to collaboratively comment and tag interesting information to share it with your co-workers.

[Go to the XWiki Watch Reader](#)

Tags

skype ebay stock Siemens, Bosch, E.ON, Innovation
realtime feed iPhone, Internet, Technik RSS, Twitter,
Feed Aggregation OpenOffice, Release

Latest flagged articles

OpenOffice 3.1 ist fertig in HeiseOnline
Rest in Peace, RSS in TechCrunchIT
eBay will Skype an die Börse bringen in HeiseOnline
eBay Plans IPO for Skype in 2010 in Mashable

Latest read articles

OpenOffice 3.1 ist fertig in HeiseOnline
Rest in Peace, RSS in TechCrunchIT
Wärmespeicher macht Solarenergie konkurrenzfähig in
WirtschaftsWocheTechnologyTrends

- This is a test comment by Alrik Degenkolb

Wie deutsche Traditionsunternehmen die
Umwelttechnik entdecken in

Keywords

Feeds

Page	Date	Last author	Actions
WirtschaftsWoche Technology Trends	2009 May 06 at 19:05	Alrik Degenkolb	Copy - Delete - Rename - Rights
TechCrunch IT	2009 May 06 at 19:05	Martin Boehringer	Copy - Delete - Rename - Rights
Mashable	2009 May 06 at 19:05	Martin Boehringer	Copy - Delete - Rename - Rights
Google Search "Enterprise RSS"	2009 May 06 at 19:05	Martin Boehringer	Copy - Delete - Rename - Rights

Figure 2: XWiki Watch overview page

The first page of XWiki Watch presents an overview of the current states and recent activities. This includes a list of currently watched feeds, a list of the last read (and commented on) as well as flagged articles, used keywords and finally a tag cloud resembling the issued tags – more details about that will follow. How all this looks in an assembled view can be seen in Figure 2 above.

From the welcome page one can navigate to the main part of the application – the XWiki Watch Reader. As can be seen in Figure 3, the focus of the main view is clearly the list of articles/items that can be scrolled through via “next” and “previous” buttons. From the perspective of a user this has the great disadvantage that she has to click from page to page to read all the news instead of smoothly scrolling the whole list in one screen. From the technical perspective, it is easier to implement it that way instead of having to load all the items in advance just to fill up a list. On the other hand, it would be possible through Ajax to load the items subsequently while scrolling, i.e. when the user pauses or stops scrolling, because otherwise the actual content is not readable anyway.

At the left side is a tree view of feeds grouped under customisable categories. Each category can hold one or more feeds and each feed can belong to one or more categories. Each group can be elapsed or contracted via the usual “+” and “-” buttons. The right side holds filters, keywords and a tag cloud.

Figure 3: XWiki Watch main page

The filters are predefined within XWiki Watch but can be selected independently of each other – they can be disabled totally or even checked all at once. One of the filters is to show only “flagged articles”. These are articles that have been marked by users of the Watch application (indicated by a little star aside the feed item). “Read articles” are (as the name suggests) the ones that have been read by any of the users. “Unread articles” are the exact opposite. “Trashed articles” are the ones that have been marked as deleted; as with the “recycle bin” known from many operating systems items aren’t deleted permanently but can be recovered for some time.

One thing that comes to mind when seeing the list of filters is probably: “How can I define my own filters?”. This is essentially performed with the keywords that define search terms. They can be imagined as custom groups showing only articles containing the specified (key-)word.

Rest in Peace, RSS   

TechCrunch IT 05/05/2009 22:04:28

Tags: RSS, Twitter, Feed Aggregation

It's time to get completely off RSS and switch to Twitter. RSS just doesn't cut it anymore. The River of News has become the East River of news, which means it's not worth swimming in if you get my drift. I haven't been in Google Reader for months. Google Reader is the dominant RSS reader. I've [...]

Comments (0)

Figure 4: Detailed posting view

Keywords are one way to navigate through the bulk of articles. However, they rely on words already present in these articles, which might not be helpful in several cases – either because some relations are not expressible by the already available words or simply because an article's short version is too short to hold every significant reference. One way out of this inconvenience is to link articles manually to “keywords” – an approach called tagging [Ma06]. As can be seen in Figure 4 above, it is possible to associate several tags with an article. Since tags are likely and meant to repeat themselves throughout several articles, it is not only possible but also useful to build an overview map of all (or at least the most used) tags. These so-called tag clouds are essentially text-based diagrams (see Figure 5) that visualise the presence and the frequency of the most used tags [Ku07]. The more often a tag is referenced, the larger its font will be and it will therefore be emphasised in relation to other tags. Since these visualisations are usually made inside a fixed (rectangular) shape with varying intensity, they are called “clouds”.



Figure 5: Detailed view of a tag cloud

The greatest problem with giving them purpose in a feed reader is that they need to be assigned first. It is not possible to scan articles automatically for tags (since that would be essentially only a keyword) and it would also not make much sense to define

them inside the feed, since everyone differs in their opinion on what to tag and how to tag – an article about “OpenOffice” might be tagged with “OpenOffice” or simply “Office”, both or even something completely different, which lies in the perception of the individual deciding about the tags. Therefore, tags can only be useful if they are all issued by the same (group of) people who are also using them. That in turn means that they necessarily require manual work.

5 A Concept for an Ideal Feed Reader

XWiki Watch shows what a social approach to feed management and hence the management of the information flow could look like. Different users have access to the same information platform and their comments, tags and favourites are accessible to everyone. By using such software, the search for the right information is a collaborative task.

However, compared with other social software applications, the XWiki Watch approach is still quite basic. Possible enhancements could be a recommendation engine for unknown feeds or similar feed items (such as those known from web shops), a rating mechanism (i.e. with a number of stars instead of only one “favourite” icon), a powerful search and rich filter mechanisms (based on metadata like tags and time stamps). Furthermore, to leverage the collaborative approach, community functions as known from social networking services [KR08] would enable collaboration, especially in bigger organisations. Another great source of information could be usage data from the feed reader. Indicators like subscription count (for feeds) or coverage percentage (how many items from a feed were read by the users) could play an important role in intelligent social feed readers.

6 Conclusion

Feeds can be considered the aorta of social software. This paper presented the surprising fact that these applications themselves do not use principles of social software at all. However, these principles could enable a richer and more effective management of information flow. Therefore, we presented XWiki Watch, a nearly unknown feed reader with collaboration functionality. While its approach is promising, there are many other possible components of social feed readers. Our future research agenda includes a detailed conception of an ideal social feed reader and the implementation and evaluation of a prototype.

References

- [Ch00] Chen, J.R., Wolfe, S.R., Wragg, S.D., 2000, A distributed multi-agent system for collaborative information management and sharing, in: Proceedings of the ninth international conference on Information and knowledge management, pp. 382–388.

- [Di04] Dickerson, Ch., 2004, Is wiki under your radar?, *InfoWorld* 26(45), p. 26.
- [HT01] Ho, J., Tang, R., 2001, Towards an optimal resolution to information overload: an infomediary approach, in: *Proceedings of the 2001 international ACM SIGGROUP conference on Supporting group work*, pp. 91–96.
- [HW05] Hippner, H., Wilde, T., 2005, Social software, *Wirtschaftsinformatik* 47(6), pp. 441–444.
- [KR08] Koch, M., Richter, A., 2008, Funktionen von Social-Networking-Diensten, in: *Proceedings Multikonferenz Wirtschaftsinformatik 2008*, pp. 1239–1250.
- [Ku07] Kuo, B., Hentrich, T., Good, B.M., Wilkinson, M.D., 2007, Tag clouds for summarizing web search results, in: *Proceedings of the 16th international conference on World Wide Web*, pp. 1203–1204.
- [Le05] Lee, I., Kim, J., Kim, J., 2005, Use contexts for the mobile Internet: A longitudinal study monitoring actual use of mobile Internet services, *International Journal of Human–Computer Interaction* 18(3), pp. 269–292.
- [Ma06] Marlow, C., Naaman, M., Boyd, D., Davis, M., 2006, HT06, Tagging Paper, Taxonomy, Flickr, Academic Article, ToRead, in: *Proceedings of the seventeenth conference on Hypertext and hypermedia*, pp. 31–40.
- [Na04] Nardi, B.A., Schiano, D.J., Gumbrecht, M., 2004, Blogging as social activity, or, would you let 900 million people read your diary?, in: *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pp. 222–231.
- [Ne99] Netscape, 1999, My Netscape Network – Quick Start, [web.archive.org, http://my.netscape.com/publish/help/futures.html](http://my.netscape.com/publish/help/futures.html).
- [Pa08] Park, J., Fukuhara, T., Ohmukai, I., Takeda, H., Lee, S., 2008, Web content summarization using social bookmarks: a new approach for social summarization, in: *Proceeding of the 10th ACM workshop on Web information and data management*, pp. 103–110.
- [Ru08] Ruby, S., 2008, RSS 2.0 and Atom 1.0 Compared, Wiki. Atom Wiki, October 9, 2008, <http://www.intertwingly.net/wiki/pie/Rss20AndAtom10Compared>.
- [Tr03] Trott, B., 2003, Why We Need Echo, Weblog, Six Apart News & Events, June 29, 2003, www.sixapart.com/about/news/2003/06/why_we_need_ech.html.
- [Ul08] Ullrich, C., Borau, K., Luo, H., Tan, X., Shen, L., Shen, R., 2008, Why Web 2.0 is good for learning and for research: principles and prototypes, in: *Proceeding of the 17th international conference on World Wide Web*, pp. 705–714.
- [Wi03] Winer, D., 2003, RSS 2.0 Specification, Harvard Law, July 15, 2003, <http://cyber.law.harvard.edu/rss/rss.html>.