EduFeedr: following and supporting learners in open blog-based courses

Hans Põldoja

Institute of Informatics, Tallinn University.

Abstract

In recent years several educators have organized open courses where participants reflect on their personal blogs. With a large number of participants it becomes a challenge to follow all the course discussions. In this paper we present the EduFeedr system that is specifically designed for following and supporting student activities in blog-based courses.

Keywords

massive open online courses, personal learning environments, research-based design, web syndication

Introduction

The open education movement has been largely focused on the openly available educational content. This work has resulted in several OpenCourseWare initiatives and open educational resource repositories. However, learning is a social process and open content is not the only way to change the educational system towards openness. In addition to open content we need open learning environments and teaching practices.

In recent years several educators have made their courses available with an open enrollment. One of the first examples was Introduction to Open Education course that was organized by David Wiley in fall 2007 (OpenContent Wiki, 2007). Course syllabus and weekly assignments were published on a wiki page. The participants were expected to write blog posts based on the weekly readings and assignments. Anybody was able to join the course simply by editing the course wiki and adding their name and blog address. This course received positive feedback from the participants (Fini et al, 2008) and acted as a model for later open courses.

In spring term 2008 the author of this paper was a co-facilitator on a similar course in University of Art and Design Helsinki. On that course the facilitators used a course blog in addition to wiki pages in Wikiversity (Leinonen et al, 2009). Wiki served as a course outline while more detailed guidelines about weekly assignments were posted to the course blog. Following all the participants and managing the learning environment was quite a challenge since more than 70 people signed up for the course. Simple tasks like setting up a blogroll and subscribing the RSS feeds required a lot of manual work. Also the participants found it complicated to follow a large number of blogs. In later courses the author has used online feed readers (e.g. Pageflakes, Netvibes) to create shared pages that contain blog posts and comments from all participant blogs. However, these solutions do not scale well with a large number of participants.

A number of people who enroll to the course may grow rapidly if the course receives attention in the blogosphere. One of the most well known open courses, Connectivism and Connective Knowledge 2008, attracted about 2200 participants (Downes, 2010). The participants of the course started to call the course massive open online course (MOOC) (Siemens, 2008).

In this paper we present the EduFeedr system that is specifically designed for following and supporting student activities in blog-based courses (EduFeedr, 2010). This article will start by presenting the challenges that we are facing when aggregating course activities from learner blogs. The paper will continue by outlining some of the related works. The fourth section is about the design methodology that was used in the project. The fifth section describes the current implementation of EduFeedr. Finally we will discuss the planned evaluation of the system and future work.

1. Design challenges

There are several aspects that make it difficult to follow student activities in massive open online courses. The number of participants is not only large but it is also constantly changing. People continue to join the course after it has started and many participants do not complete the course. Therefore it is complicated to have an up-to-date list of active participants.

It is common, that course activities take place not only in blogs but also in other Web 2.0 environments (Twitter, Delicious, etc.). Learning resources may be also published in several locations depending on the type of resource (presentations in SlideShare, videos in YouTube, etc). This information could be aggregated if participants have agreed on the course tag that they will use in other systems.

Feed readers display only updates from active users. In a course setting it is also important to notice, when some participants haven't posted anything for a while. Getting comments and feedback may motivate them to continue with the course. Current feed readers do not take into use the social network data that comes from links and comments between the blogs.

There is no centralized way to see all the comments that a person has made in other blogs. It is common that commenting other participants' blog posts is one of the assessment criteria in the courses where learners use blogs. Also it might be difficult to identify who has written the comment, because people do not have to log in to write comments.

Finally, the information in this kind of learning environment may disappear quickly. We have learned that some people make their blogs private after they have completed the course. In order to aggregate the blog posts they should be available in the RSS feed. These feeds contain only most recent blog posts. In a similar way Twitter shows only most recent tweets when they are searched by a hashtag.

In EduFeedr project we are trying to address these challenges by developing an online feed reader that could be used for enrolling to the course and following the course activities that take place in a distributed learning environment.

2. Related works

The use of technological tools in massive open online courses has been studied by Fini (2008). He analyzed learners' attitudes towards the tools that were used in the Connectivism and Connective Knowledge 2008 course. This course had a quite diverse learning environment where Moodle, Google Groups, Twitter and several other tools were used in addition to blogs. According to this study a shared feed reader Pageflakes was considered the least useful tool.

Stephen Downes has developed a gRSShopper newsletter software that is used in the Connectivism and Connective Knowledge course and his other open courses (Downes, 2010). The participants have to submit addresses of their blogs and the system will aggregate all the blog posts that contain the course tag. The facilitator will send out a daily newsletter that contains summaries of selected blog posts and Twitter tweets with the course hashtag.

There have been attempts to add blog aggregation to institutional learning management systems (LMS's). One of these is BAM that is integrated with Webfuse LMS (Jones, 2009). A newer version named BIM is a blog aggregation module for Moodle (Jones, 2010).

P2PU has customized Drupal content management system to provide open courses on a centralized platform (P2PU, 2010). However, their courses have typically a limited number of participants and they do not have to deal with hundreds of people in a same course.

3. Design methodology

The design process of EduFeedr is based on the research-based design methodology (Leinonen et al., 2008). We have used this methodology in earlier projects (Leinonen et al. 2010) and it has proven to be a flexible and lightweight methodology. The research-based design process is divided into four iterative stages, which may take place partly in parallel: (1) contextual inquiry, (2) participatory design, (3) product design, and (4) software prototype as hypothesis.

The aim of contextual inquiry phase is to define the context and preliminary design challenges. In the case of EduFeedr the context is open online courses where all participants use their personal blogs. The design challenge is to present the learning activities that take place in blogs and other Web 2.0 environments in a compendious and user-friendly way. In the contextual inquiry phase the authors of EduFeedr organized 4 blog-based courses where they used various online feed readers to aggregate blog posts. The main limitations of current feed readers came out during these courses.

The second phase of research-based design is a participatory design. Participatory design approach emphasizes the importance of involving all the stakeholders from the early phases of the design process. The main method that we used in this phase was scenario-based design (Carroll, 2000). The designers wrote six scenarios that described how teachers and students used an educationally enhanced feed reader named EduFeedr. These scenarios included (1) first experience with EduFeedr, (2) posting a response to the assignment on the student blog, (3) exploring the connections between student blogs, (4) setting up course feeds, (5) archiving course posts and comments, and (6) using the offline client.

These scenarios were initially presented in the Open Education course organized by Mozilla Foundation, ccLearn and Peer 2 Peer University. A set of questions was prepared about each scenario and it was possible to comment the scenarios in a wiki. Later two participatory design sessions were organized with people who have been teaching in blog-based courses. In these sessions the participants read the scenarios and we had a structured discussion based on the questions about each scenario. The participants agreed with most of the scenarios but they were quite critical about some details. For example it came out that they do not expect that most of their students will start adding tags to their blog posts.

Currently we are between the third and the fourth phase of research-based design. The aim of a product design phase is to define the use cases and basic interaction with the system. Two main methods that we have used in this phase are user stories (Cohn, 2004) and paper prototyping (Snyder, 2003). Each user stories describes one feature or requirement in two or three simple sentences. User stories are published in the software development environment Trac where it is possible to discuss and revise them.

The final phase of research-based design is software prototype as hypothesis. Our hypothesis with EduFeedr is that distributed learning environments need a central component that will collect, combine and present the learning activities that take place in learners' personal learning environments. Early prototypes were tested internally using data from old courses. The first public prototype of EduFeedr was made available in September 2010.

4. Description of EduFeedr

Currently we have implemented the basic features of EduFeedr that include setting up the course, enrolling to the course, aggregating blog posts and comments, and visualizing the progress and social network.

EduFeedr is designed so that only facilitator needs a user account in the system to set up the course. Anybody can view the aggregated blog posts and comments without logging in. When setting up the course it is important to specify the location of the course blog where all the assignments will be published.

Participants can enroll to the course by filling up a simple form with their name, e-mail and blog address. In the current implementation we support only blogs that run on Blogger or WordPress. We have this limitation because the system must also detect the location of the comments feed. It is possible to specify a deadline for the enrollment. After that time only the facilitator can add people to the course.

Each course is divided into six sections: (1) course feed, (2) course info, (3) participants, (4) assignments, (5) progress, and (6) social network. The first section is a Course feed that displays 10 most recent blog posts and comments from the course (see Figure 1). The facilitator can hide those posts and comments that are not related with the course.

In the participants page it is possible to see a list of participants who have enrolled to the course. If people prefer to use their own feed reader for following the course they can download OPML files that contain all the blog post feeds and all the comment feeds. The facilitator can also download a vCard file with e-mails of all participants. This file can be imported to the address book. EduFeedr provides also an HTML code that contains blogroll with all the participants. This can be copied to a widget in the blog sidebar.

In the assignments page the facilitator has to specify these blog posts in the course blog that contain assignments. When participants write their responses to the assignments their blog posts should contain a link to the assignment in the course blog. Based on these links we can display a progress diagram that shows how many assignments the participants have completed. An example progress diagram is on Figure 2. From the progress diagram it is possible to access individual blog posts. It means that it is possible to read the course blogs without leaving EduFeedr. Only comments have to be written in participants' blogs.

The last section has a social network diagram that shows connections between the participants. A connection is made when one links or comments another participants' blog post. The comments are connected with the course participants based on the URL that was submitted in the comment metadata. The social network data can be also downloaded in tab separated format. This can be used in external visualization services such as ManyEyes by IBM.

Technically EduFeedr is built as a plug-in for Elgg social networking engine. We decided to use Elgg because we have previous experience with the platform (Sillaots & Laanpere, 2009). EduFeedr works as a frontend, that displays aggregated course data from local database. Aggregating the blog posts and comments is handled by a separate web service component named EduSuckr. We have also used various open source libraries such as SimplePie, JSViz and NuSOAP. The source code of EduFeedr is available under the GNU General Public License v2.

5. Conclusions and future work

This paper described a work that is still in progress. We have recently launched the public beta version of EduFeedr. In autumn term 2010 EduFeedr will be evaluated in several courses in Tallinn University and Estonian Wikiversity. The first one of these courses has already started. The aim of this evaluation is to find out how learners use EduFeedr and which problems occur. In addition to these courses we are planning to organize usability testing with a small group of users using think aloud protocol.

Our current courses have typically only 20–30 participants. In the development phase we used data from the course that started with 70 participants. Testing EduFeedr on a truly massive open course would be an interesting challenge.

We are going to continue the development of new features after fixing the found defects and analyzing the user feedback from the first courses. Some plans for future developments include (1) aggregating recent content from various Web 2.0 services that are used in courses (Twitter, Delicious, SlideShare, etc.), (2) providing visualization widgets that could be embedded in external web sites and (3) archiving the course posts and comments.
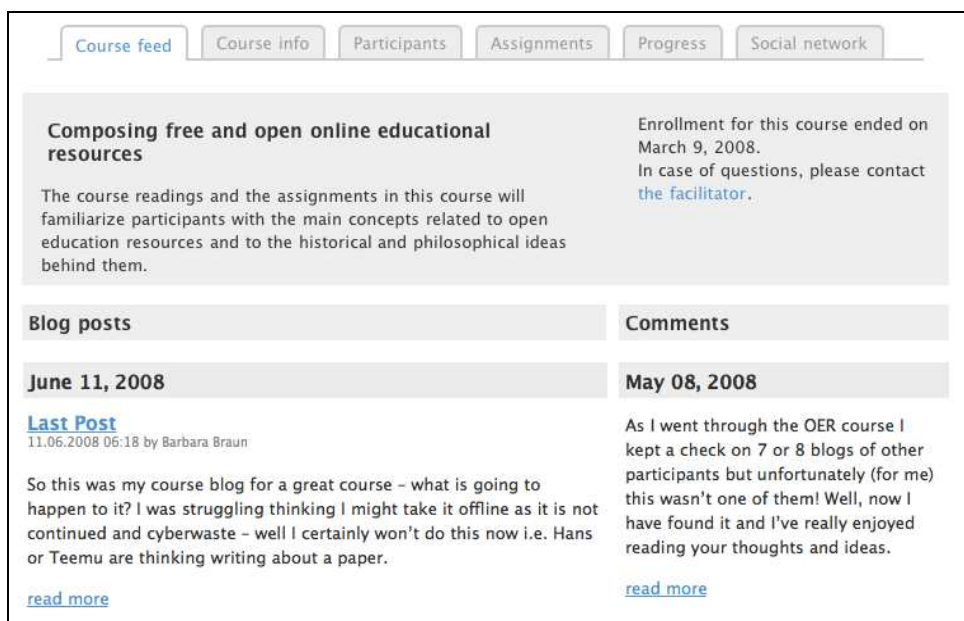
6. Acknowledgements

Figures

Figure 1. Course feed displays recent blog posts and comments

Figure 2. Progress page displays participants' progress

Bibliographic references

Carroll, J.M. (2000). *Making Use: Scenario-Based Design of Human-Computer Interactions.* Massachusetts: The MIT Press.

Cohn, M. (2004). *User Stories Applied: For Agile Software Development.* Boston: Addison-Wesley.

Downes, S. (2010). New Technology Supporting Informal Learning. *Journal of Emerging Technologies in Web Intelligence, 2*(1), 27–33.

EduFeedr. (2010). Retrieved from http://www.edufeedr.net/

Fini, A., Formiconi, A., Giorni, A, Pirruccello, N.S., Spadavecchia, E., & Zibordi, E. (2008). IntroOpenEd 2007: an experience on Open Education by a virtual community of teachers. *Journal of e-Learning and Knowledge Society, 4*(1), 231–239.

Fini, A. (2010). The Technological Dimension of a Massive Open Online Course: The Case of the CCK08 Course Tools. *International Review of Research in Open and Distance Learning, 10*(5).

Jones, D., Luck, J. (2009). Blog Aggregation Management: Reducing the Aggravation of Managing Student Blogging. In G. Siemens, & C. Fulford. (Eds.). *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2009* (pp. 398–406). Chesapeake, VA: AACE.

Jones, D. (2010). BIM – Feed Aggregation Management and Marking. Retrieved from http://davidtjones.wordpress.com/research/bam-blog-aggregation-management/

Leinonen, T., Toikkanen, T, & Silfvast, K. (2008). Software as Hypothesis: Research-Based Design Methodology. In *Proceedings of Participatory Design Conference 2008, Indiana University, Oct 1–4 2008.*

Leinonen, T., Vadén, T., & Suoranta, J. (2009). Learning in and with an open wiki project: Wikiversity's potential in global capacity building. *First Monday, 14*(2). Retrieved from http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/art icle/view/2252/2093

Leinonen, T., Purma, J., Põldoja, H., & Toikkanen, T. (2010). Information Architecture and Design Solutions Scaffolding Authoring of Open Educational Resources. *IEEE Transactions on Learning Technologies. 3*(2), 116–128.

OpenContent Wiki. (2007). Intro Open Ed Syllabus. Retrieved from http://www.opencontent.org/wiki/index.php?title=Intro_Open_Ed_Syllabus

P2PU (2010). P2PU. Retrieved from http://p2pu.org/

Siemens, G. (2008). MOOC or Mega-Connectivism Course. Retrieved from http://ltc.umanitoba.ca/connectivism/?p=53

Sillaots, M., Laanpere, M. (2009). Building the Next-Generation Educational Portal for Estonian Schools. In *The European School 2.0: The Seventh Open Classroom Conference, Porto 15.–17. October 2009*. Porto: EDEN.

Snyder, C. (2003). *Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces.* San Francisco: Morgan Kaufmann.

About the author

Hans Põldoja

Institute of Informatics, Tallinn University

Hans Põldoja is a designer and a research associate in the Institute of Informatics at the Tallinn University (Estonia) and a doctoral candidate in the Media Lab Helsinki at the Aalto University School of Art and Design (Finland). His main research interest is the conceptual design of open and personal learning environments. His most noteworthy project is the open educational resources authoring and sharing community LeMill. Hans blogs at www.hanspoldoja.net.

Institute of Informatics, Tallinn University
Narva mnt 25, 10120 Tallinn, Estonia
hans.poldoja@tlu.ee