

Integrating industrial partners into e-teaching efforts

Using social networks to support the initiation of co-operations

Tobias Fries, Sebastian Boosz, Andreas Henrich

University of Bamberg, Chair for Media Informatics
Feldkirchenstraße 21, D-96045 Bamberg
tobias.fries@uni-bamberg.de, sebastian-wilhelm.boosz@stud.uni-bamberg.de, andreas.henrich@uni-bamberg.de

Abstract

Since 2008 we have been using the existing university-wide learning management system (LMS) “Virtual Campus” (VC) to incorporate industrial partners into e-teaching courses at the University of Bamberg. We have developed different co-operation types and evaluated their appropriateness in pilot co-operations between lecturers of the University of Bamberg and regional industrial partners—respecting legal requirements, too. The vast amount of LMS courses available for students lead to a special co-operation web portal for companies, which we have ported to common social networks using the OpenSocial API. The social network application elevates the convenience for the company users, supports the matching of companies’ interests and co-operation topics and uses the social graph of the project partners to reach potential attendees.

In this paper we give a general overview of popular social networks. We describe similarities and differences in target groups, APIs, and also application and data protection policies. We explicate the Use Cases of our application and explain the process of developing applications based on the OpenSocial API. We also describe the experiences we have made with the application admission procedures of different social networks and give an outlook on how we want to utilize Facebook’s Social Graph API and embed Facebook widgets and features (e.g. recommendation and authorization) into the existing web version of our co-operation portal.

1 Background

Centrally administered university-wide learning management systems (LMS)—which have evolved out of the former variety of competing LMS used by different faculties or chairs at most universities in recent years (Henrich, Sieber, Wolf 2007)—provide a powerful infrastructure to support co-operations (co-ops) and bi-directional knowledge transfer between universities and companies. They help to overcome obstacles of temporal synchronization and spatial distances in co-ops between different institutions.

Fig. 1 describes the usage scenario to incorporate company management and staff into academic teaching. The green arrows symbolize the conventional usage of a university LMS: Lecturers provide topics and content to students and setup discussion forums about course topics. Students mainly access and download provided content. Forums and chats are used for communication between lecturers and students. Our approach incorporates two more parties: company management and staff. Company staff can access course content and—especially in exercise courses or seminars—the management can provide topics and content to increase practical relevance of the courses. The students should also be encouraged to upload content and all four parties can discuss topics via the LMS (red arrows). Our approach permits different types of co-ops dependent on the kind of course (lecture, seminar or exercise course) and the desired degree of involvement (active/passive) of the companies. At the University of Bamberg we had co-ops ranging from granting company members access to lecture notes, slides, lecture recordings and discussion forums up to courses, where students tested, evaluated or improved company products. Company representatives are not just learners, but contribute knowledge, experience and real-life questions to the courses.

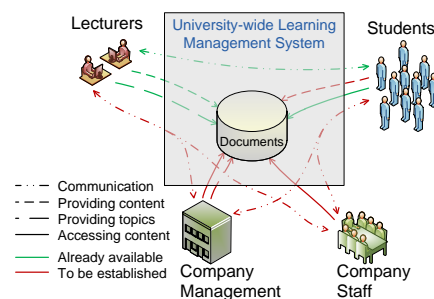


Fig. 1: Incorporating companies

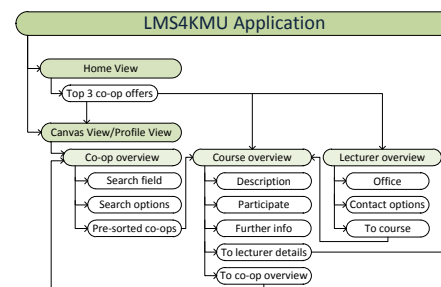


Fig. 2: Views of our app

When granting company members access to certain courses in a university LMS, privacy, data protection and copyright issues have to be considered, but common LMS offer built-in or additional features to support the legal requirements (Fries, Henrich 2010b). Another critical success factor is minimizing necessary efforts for companies and lecturers to start co-ops. For that purpose we have developed a co-op web portal that provides access to only the courses available for companies (out of over 1,400 courses in the LMS “Virtual Campus”¹ (VC) in winter 2010/11). The portal harvests information about courses and lecturers from different IT systems in use at the university and provides them in a target-group specific manner to the companies (Fries, Henrich 2010a).

The co-op web portal² is a convenient tool for the company members to find suitable co-op offers, get contact information about the lecturers and start co-ops. Our industrial project partners experienced no major problems in using the web portal. After the successful pilot phase of our project, we want to invite further companies to co-operate with lecturers and students. The key to this is getting company members to navigate to our co-op portal. The usage of social networks (SNs) like Xing³, LinkedIn⁴ or Facebook⁵ (FB) has increased massively and their APIs have become very powerful and standardized (apart from FB almost all SNs provide Google’s OpenSocial (OS) API⁶). So, integrating our co-op portal into SNs and integrating SN widgets into our web portal may have potential to further increase the convenience for interested companies and to recommend our portal to their contacts.

In chapter 2 we give a brief overview of popular SNs. We identify major similarities, show main differences and compare their APIs. Chapter 3 presents use cases (both adapted from our web portal and newly designed ones) of SNs for our purpose including carefully accessing users’ data. In chapter 4 we describe our app prototype and experiences with the networks and APIs. Chapter 5 gives an outlook on future work.

¹ <http://vc.uni-bamberg.de/moodle> (checked Jan 17, 2011)

² <http://www.uni-bamberg.de/minf/lms4kmu-portal> (checked Jan 17, 2011)

³ <http://www.xing.com> (checked Jan 17, 2011)

⁴ <http://www.linkedin.com> (checked Jan 17, 2011)

⁵ <http://www.facebook.com> (checked Jan 17, 2011)

⁶ <http://code.google.com/intl/de-DE/apis/opensocial> (checked Jan 17, 2011)

2 Social Networks, APIs and Application Quality

With the increasing popularity of SNs, a change in the way of how people search information on the internet may be about to take place. While google.com is still the most visited website with a daily reach of approx. 45% of global internet users, facebook.com has reached the second rank with about 39%⁷. Instead of using search engines to find information about a more or less clearly defined topic, many people browse information recommended by their friends over SNs. Microsoft and FB seem to believe, that both approaches can be reasonably combined to subjectively increase quality of search results, as Microsoft's Bing will soon display recommendations of the searcher's FB contacts among the usual search results (Braun, 2010).

With over 500 million registered users, FB is the most popular global SN. Started as a SN for students at the Harvard University in 2004 and extended to other US students in 2006, it is now available in over 80 languages for everyone. In Germany a very similar SN called studiVZ⁸ (VZ) was started. In the beginning, when no German FB localization existed, VZ was clearly the most popular SN amongst German students. But in recent years, FB has become more and more popular amongst German students. VZ has also opened up for non-students, providing SNs called meinVZ and schülerVZ which are closely connected to VZ. FB and VZ have a clear focus on leisure time and entertainment. For both platforms a vast amount of third party apps (mainly games) exist.

Despite the general dominance of FB (over 14 million registered German users⁹) and the VZ networks (over 17 million users¹⁰) niche markets for specialized SNs still exist. For our attempt to initiate co-ops with companies, especially SNs with a more business-oriented focus are worth examining. In Germany, Xing is the most popular amongst these, whereas globally, LinkedIn has the most registered users in this domain (Briegleb, 2010). Both platforms provide third party apps, but in contrast to the thousands of mainly entertainment focused apps available in FB and VZ, Xing and LinkedIn allow only a handful of apps.

⁷ <http://www.alexa.com/topsites> (checked Jan 17, 2011)

⁸ <http://www.studivz.net> (checked Jan 17, 2011)

⁹ <http://www.facebakers.com/facebook-statistics/germany> (checked Jan 17, 2011)

¹⁰ http://www.studivz.net/l/about_us/1 (checked Jan 17, 2011)

APIs and third-party apps In May 2007 FB opened the website for third-party apps by launching the so called “FB Platform”. Apps are displayed in a container inside a FB page and can include FB widgets (e.g. the Like Button) and access users’ profile data. The app can be implemented as usual JavaScript-enhanced HTML page (iframe) or in the FB Markup Language (FBML) and FB JavaScript (FBJS). To interact with FB’s backend, a REST API is provided and user data can be accessed via the FB Query Language (FQL) (Graham, 2008). In December 2008 “FB Connect” became available, allowing developers to include FB’s widgets into independent websites not running as apps inside FB. In 2010 the REST API was replaced by the more modern Graph API, giving developers three possibilities for interaction with the SN: (1) *FB for Websites* allows the usage of social plugins and comfortable solutions like a single sign-on procedure on any website. (2) *Apps on Facebook.com* allows to run web apps in a dedicated iframe on FB, place widgets on users’ profile pages and update a user’s FB stream. (3) *Mobile Apps* is a collection of services for web apps running on mobile devices (Krutisch 2010).

In contrast to the proprietary Social Graph API which is limited to FB, Google released version 0.7 of its counterpart OS API in late 2007 under an Apache 2.0 and Creative Commons license. Most SNs have implemented the OS API so far¹¹. So Google’s aim to provide an API for cross-platform apps seems to be achieved, although different versions of the API are implemented in the SNs (version 1.1 was released in November 2010, but version 0.8 (Wartala 2008) is still very wide-spread). OS can be seen as a container for apps written in HTML, JavaScript and CSS, which can be placed on web pages inside SNs, providing a standardized AJAX request mechanism to contact the app developer’s web server and also standardized access to user data in the SN (Häsel, Iacono 2010). For security and data protection reasons, SN users can define access rules for apps in general or individually when they add an app to their profile page (cf. chapter 4).

As Xing is the most popular business network in Germany, we started implementing our app as an OS 0.8 app, focusing on Xing’s requirements. Xing supports three OS views, Home View (HV), Canvas View (CV) and Profile View (PV). The HV is an app container to be included in the user’s

¹¹ <http://wiki.opensocial.org/index.php?title=Containers> (checked Jan 17, 2011)

personal page with a fixed size of 315 x 180 pixels. It is only available for users who added the app to their accounts. HV can access user data if permitted by the user. CV can be larger than HV and therefore display more content. Like HV it is only accessible for the user himself and can access the same data as HV. PV can be as large as CV and is displayed inside the user's profile page, which is available to other users depending on the user's permissions. Apart from the owner's data, PV can also access data of the user viewing the app on another user's profile page (if the viewer permits).

3 Use Cases

The prototypical implementation of our app on iGoogle¹² demonstrates number 1 to 6 of the Use Cases described on page 7 and depicted in fig. 3 on page 8. The app provides the functionality of our web portal (Fries, Henrich 2010a) and has additional advantages: (1) The project partners (companies and lecturers) are already using Xing to manage their business contacts; our app valorizes the well-known environment. (2) Accessing users' interests via OS, co-op offers can be ordered and filtered accordingly. (3) Users' friends are informed about the app when browsing status updates or visiting profile pages. (4) Users can easily recommend the app within the SN. (5) The SN profile can be used to identify and authorize users for personalized features of the web portal. (6) Lecturers, companies and students can communicate via the SN with others that have added the app.

The OS version of our app supports three in most SNs available views (cf. chapter 2). The HV with very limited display size provides the three most appropriate co-op offers for the user. It is placed on the homepage of the user that has installed the app when he logs in to the SN. The CV and PV of the app provide the same functionality as our web portal, but the co-op offers are ordered by relevance for the user who has installed the app (CV) or who views the app (PV). Fig. 2 on page 2 depicts the relationship between the views. When the user logs in to the network, he finds the HV, containing the three co-op offers that best match his interests. If he clicks on "more co-op offers" the larger CV is displayed, containing all available co-op offers

¹² <http://www.google.de/ig/directory?q=lms4kmu> (checked Jan 17, 2011)

ordered by relevance for his interests, a basic search field and advanced search and filter options. If the user clicks on a co-op offer in HV or CV, a course overview is displayed in CV, giving a description, as well as links for participation, further information and the lecturer's contact data. If the user clicks on a lecturer's name or picture (HV) or on the "lecturer's details" link (course overview in CV), the lecturer's contact data are displayed in CV. If a user that has not installed the app (yet) finds it on another users profile or gets a recommendation, he can access the PV, which is similar to the CV.

Fig. 3 on page 8 depicts the eight Use Cases (UC) our OS app is designed for. UCs 1 to 6 have already been implemented in our prototype, UCs 7 and 8 are to be implemented next. **UC 1** allows to look for an interesting co-op offer. Having logged in, the user finds the top three co-ops for his interests in the app's HV on his Xing homepage and can click on an offer to get more information about it or click on "see more offers" to get the full list (in both cases the CV is being displayed). **UC 2** starts with a Xing user finding the app (by reading in the status updates that a friend has installed the app or joined a co-op, watching another user's profile, getting a recommendation from another user or browsing the list of available apps). When he opens the app, the PV is displayed, showing all available offers (ordered by his interests if permitted). He can use most of the features of the portal and install the app to get full access. **UC 3** is about searching for co-ops. The user can just type in a keyword to the search field (CV or PV) and the app will display matching co-ops only. He can also use the advanced search where he can define, which fields (co-op title, description, information for companies, lecturer name) should be included in the search and which criteria (current semester, co-op type, lecture type) the possible co-ops have to fulfill. **UC 4:** When the user has found an interesting-looking co-op in any view, he can get more details about it by clicking on the co-op name. When he gets a recommendation, clicking the recommendation link also displays the co-op details (after the user has installed the app). The view changes to CV in any case. **UC 5** allows contacting the lecturer for a co-op. It is possible to click on the lecturer's name next to a co-op offer in any of the three Views (HV also provides a clickable picture) or click on "contact data" (course overview in CV or PV). If the lecturer is present at Xing himself, the user is forwarded to his Xing profile page showing his current contact data. If the lecturer does not have a Xing profile, the app shows the lecturer's picture, office address, university webpage, email address and telephone number in CV or PV.

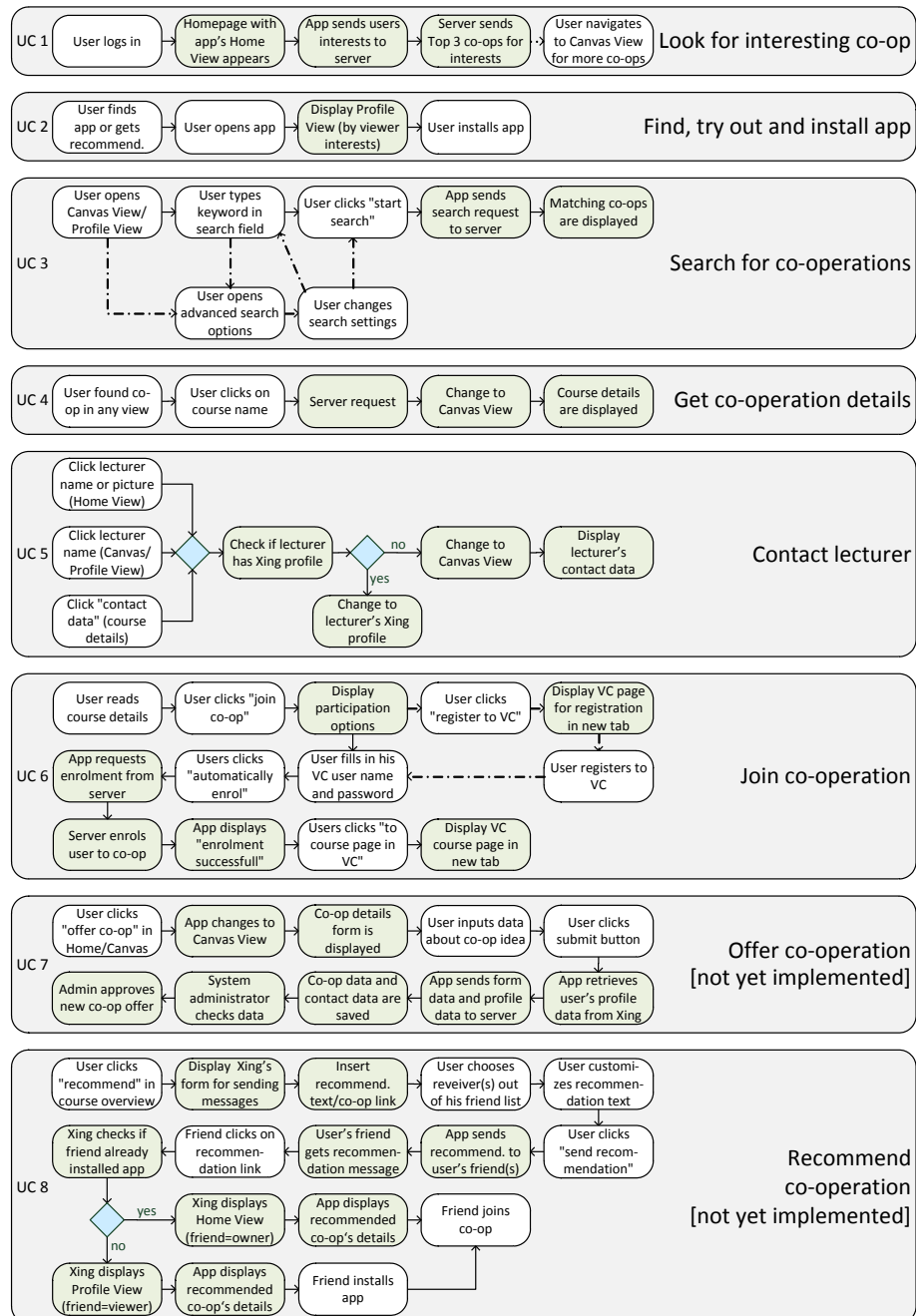


Fig. 3: Use Cases of our OpenSocial application

UC 6: When the user has decided to join a co-op, he can click on the corresponding link (in course overview). If he declares not to have a user account for the university LMS yet, the self-registration page for the LMS opens in a new browser tab. As soon as the user has registered to the LMS, he can enter his user name and password and “auto-enrol”. The app enrolls him to the course and provides a link for direct access to the course page.

UCs 1 to 6 are derived from the UCs of our web portal and are also available for users/viewers that do not allow the app to access their profile data. UC 7 and 8 explicitly require access to the user’s profile. **UC 7:** Company members can offer co-op ideas to lecturers by entering their idea and the possible co-op type in a form in CV. The app sends these together with the user’s contact data to our server, where they are checked and added to the available co-ops. Also lecturers can add ideas about future courses, mutually-planned with companies, to the portal. **UC 8** provides a convenient possibility to recommend co-ops between users of the SN. When a user finds a co-op offer that could be interesting for a friend, he can just click on “recommend” (course overview). The app will then open a standard message sending dialogue of the SN and insert the subject, a default recommendation text and a direct link to the co-op offer to the message. The user can then choose the receiver(s) of the message from his friends list and change the text or subject. The friend(s) receive the message and can click on the link. If the friend has already installed the app, the CV opens, displaying the course overview. If the friend has not installed the app yet, the senders PV of the app displays the course details and the friend can install the app.

4 Implementation and Experiences

We have developed an OpenSocial app for Xing first, because its target audience highly resembles ours. As Xing does not allow general access to the developer’s sandbox, we started developing and testing our app using the OS Development Environment¹³ (OSDE) for Eclipse, based on Apache Shindig¹⁴ (Häsel, Rieke 2009). Apart from a required XML file, specifying metadata and views (Grewe, 2009), we have implemented the app completely in

¹³ <http://code.google.com/p/opensocial-development-environment> (checked Jan 17, 2011)

¹⁴ <http://shindig.apache.org> (checked Jan 17, 2011)

JavaScript. Most of the code from the existing web portal could be adopted in a straightforward way. Functions handling AJAX requests had to be adapted to work with OS's AJAX mechanism, functions accessing users' profile data had to be implemented and the app had to meet Xing's corporate design. Apart from our Shindig server, we also tested our app on iGoogle, where it is still available¹⁵. Unfortunately, Xing rejected our concept and app prototype immediately, possibly due to restrictive app permission policies.

We then offered our app to LinkedIn, which has a similar target group as Xing, but a more international focus. In contrast to Xing, LinkedIn does not ask for a detailed concept paper, but offers a web form, where one can briefly describe the app idea¹⁶. Unfortunately, we have only received an automatic reply, but no decision about acceptance or rejection of our app from LinkedIn so far. At present we concentrate on VZ and FB, which are not as business oriented, but reach a lot of people and have less strict demands for third party apps. On both platforms getting access to the developer's sandbox is straightforward and allows testing the app directly within the SN.

We started developing the FB version as "App on Facebook.com"—rendered on our own server and integrated into FB as iframe. Just entering the URL of our existing web portal was sufficient for the start. Accessing the user's profile data and integrating FB widgets ("Like" button) are the upcoming targets. On VZ, the previously developed OS version works in principle, however VZ supports different views than Xing and has different size restrictions. Also some of our JavaScript DOM manipulations seem to be overwritten by VZ's CSS styles.

Privacy policies The SNs follow different approaches to protect their users' data: Xing apps can only access data declared to be public by the user and the contact list of the user who installed the app. Data of viewers, not having installed the app, are protected. VZ uses a system of virtual business cards which the users create and allocate to their installed apps. Besides basic information, the app may only access the information given on each vcard or if necessary ask the user to update his vcard. Accessing viewers' data is not possible. Additionally, Xing and VZ demand links to the general business

¹⁵ <http://www.google.de/ig/directory?q=lms4kmu> (checked Jan 17, 2011)

¹⁶ <http://developer.linkedin.com/themes/linkedin-home/form-inapps.html> (Jan 17, 2011)

terms and to the privacy policy of the app provider in each rendered view. On FB all apps can access general information like the user's and his friends' names, the profile picture URL and his unique user ID as well as all information with visibility level "Everyone". Furthermore the user can define privacy settings applying to all apps and different settings for individual apps. FB was recently criticized for allowing apps to access viewers' data by default (Lischka, 2010). To be trustworthy, our app only requests those fields of user data absolutely necessary to provide its functionality, and basic features of the app are available for users, who do not allow profile access at all. It will also not send messages to the user's friends unless he provokes.

5 Outlook

SNs provide handy interfaces for third party developers to utilize users' social graphs and profile data to increase an app's range and usability. Most SNs provide Google's OS API, allowing the development of platform-independent apps, however the most popular SN FB requires an individual implementation. Business-oriented SNs like Xing and LinkedIn provide a more homogenous and less entertainment-oriented user group than FB and VZ, but have stricter app acceptance policies. Our prototypical OS implementation fulfills its purpose and works fine on iGoogle and in the sandbox of VZ. The existing web portal can be included into FB as iframe. We will enhance it with FB widgets and access to the social graph. So we will be able to implement two advanced Use Cases in the OS and FB app version as well as in our web portal version (using "FB for Websites").

To reach new users we also want to provide apps for smartphones. A slightly modified version of our web portal¹⁷ already works on Adobe Air¹⁸, which has lately been released for Android 2.2¹⁹. We also plan to implement versions for iOS, Symbian and Windows Phone 7 utilizing the special features of the mobile devices (touch screen etc.) to increase convenience and user experience.

¹⁷ <http://www.tobiasfries.de/lms4kmu/lms4kmu.air> (checked Jan 17, 2011)

¹⁸ <http://www.adobe.com/de/products/air> (checked Jan 17, 2011)

¹⁹ <http://www.heise.de/newsticker/meldung/Adobe-Air-fuer-Android-verfuegbar-1104071.html> (published Oct 10, 2010)

References

- Braun, H. (2010). Microsoft und Facebook arbeiten gemeinsam an Bing. In: c't 23/2010: Heise, p. 30.
- Briegleb, V. (2010). Xing mit bisher bestem Quartalsergebnis [Electronic Version]. Heise online Newsticker. Retrieved October 14, 2010 from <http://www.heise.de/newsticker/meldung/Xing-mit-bisher-bestem-Quartalsergebnis-1054604.html>.
- Fries, T.; Henrich, A. (2010a). Integrating Industrial Partners into e-Teaching Efforts—a Portal to Support the Initiation of Co-operations. In: IADIS eLearning 2010 (IADIS International Conference e-Learning 2010). Freiburg.
- Fries, T.; Henrich, A. (2010b). Integrating Industrial Partners into e-Teaching Efforts—legal pitfalls and circumventions. In: Book of Abstracts 2010 (Online EDUCA Berlin 2010). Berlin.
- Graham, W. (2008). Facebook API Developers Guide. New York: Apress.
- Grewe, L. (2009). OpenSocial Network Programming. Indianapolis: Wiley Publishing, Inc.
- Häsel, M.; Iacono, L. (2010). Soziale Netze mit OpenSocial erweitern—Offene Beziehung. In: iX 6/2010: Heise, p. 134-140.
- Häsel, M.; Rieke, K. (2009). Aktuelles Schlagwort—OpenSocial. In: Informatik Spektrum 32/3, 2009: p. 250-254.
- Henrich, A.; Sieber, S.; Wolf, S. (2007). Integration eines hochschulweiten LMS in die Systemlandschaft einer Hochschule—ein pragmatischer Ansatz. In: Flexibel integrierbares e-Learning—Nahe Zukunft oder Utopie?: Proc. Workshop on e-Learning 2007. Leipzig: Hochschule für Technik, Wirtschaft und Kultur, p. 57–76.
- Krutisch, J. (2010). Facebook-Schnittstelle für Entwickler. In: iX 10/2010: Heise, p. 38-43.
- Lischka, K. (2010). Facebook-Anwendungen reichen unerlaubt Nutzerdaten weiter [Electronic Version]. Spiegel Online. Retrieved October 28, 2010 from <http://www.spiegel.de/netzwelt/web/0,1518,723636,00.html>.
- Wartala, R. (2008). Anwendungen mit der Opensocial-API entwickeln. In: iX 9/2008: Heise, p. 54-60.