Providing Social Sharing Functionalities in LearnWeb2.0

Ivana Marenzi, Sergej Zerr, Wolfgang Nejdl
L3S Research Center, Hannover, Germany
{marenzi, zerr, nejdl}@L3S.de

Abstract
Within the TENCompetence project we are working on an open source infrastructure for the creation, storage and exchange of learning objects and knowledge resources. We implemented LearnWeb2.0 - a prototype, which provides appropriate functionalities for the aggregation and annotation of Web 2.0 resources for lifelong competence development activities. This paper focuses on the next steps planned, describing the main functionalities to be implemented in LearnWeb2.0: resource selection, batch annotation and sharing, notification using SpreadCrumbs [4], resource aggregation using GroupMe [1] and sequencing, motivated by a knowledge sharing scenario at the University of Pavia.

1. Introduction
Web 2.0 is a challenging environment, in which knowledge resources are distributed among a set of heterogeneous online storage tools, each of which provides specific functionalities. On their desktop, people often bring together documents that share similar types, topics or proximity in time of creation [7][8], which can then be used by file management and search software to assist users in finding and aggregating resources relating to a common learning activity. However, on the web, users are often forced to distribute related resources across different Web 2.0 applications according to the type of resource, e.g. pictures in Flickr, videos in YouTube, and bookmarks in del.icio.us, even though all these resources belong to one and the same learning context [3].

To integrate models and tools for the creation, storage and exchange of knowledge resources, we are implementing the LearnWeb2.0 infrastructure [6], which makes Web 2.0 information accessible in ways that better support lifelong learning and knowledge sharing. Figure 1 depicts the architecture.

Whereas conventional Web 2.0 applications support a limited set of predefined tasks (such as storage, editing or discussing resources of specific types), our integrated LearnWeb2.0 environment aims to provide a rich set of functionalities and a homogeneous overview over the entire distributed learning space, without unnecessary boundaries.
The first LearnWeb2.0 prototype provides the basic functionalities for the aggregation and annotation of Web 2.0 resources. Figure 2 shows the user interface of the LearnWeb2.0. In this paper, we analyze and describe new important functionalities that are necessary to automate time-consuming user actions, in order to make LearnWeb2.0 more user friendly.

![LearnWeb2.0 User Interface](image)

We will use a knowledge sharing scenario at the University of Pavia to identify the required infrastructure and user level functionalities in section 2, followed by a more detailed discussion of the platform functionalities to be implemented in section 3.

2. Scenario “Knowledge Sharing at the University of Pavia”

We use a real scenario from the University of Pavia, focusing on ICT technicians working in the Information Systems Area at this University. The tasks performed by the technicians include the management of the ECDL Test Centre and the organization of learning courses. So far, these have been provided by a private company and the technicians have supported them as online-tutors or examination providers. Recently a new task has been assigned to the group: the creation of two new e-learning courses related to ECDL materials (Advanced Access and PowerPoint courses). For these topics, many related materials are already available on the Internet, so we need to support proper organization of the work (searching for and organizing resources, as well as sharing them in the group) as well as detailed planning (implementation and delivery). The intention is to use Moodle as a platform to deliver the final content. Each technician will be responsible for a specific task: Gabriele will search for appropriate resources, Daniela will provide evaluation exercises, Ivana will organize the contents in keeping with the AICA Syllabus. External contributions are also welcome: Luigi, who works on other tasks in Cremona, supports the group’s work, monitoring partner activities.

In the following subsections we illustrate the course construction process focusing on three main aspects: Searching for resources, sharing search results and aggregation and sequencing, connecting our knowledge sharing scenario with the appropriate functionalities described in section 3.

2.1 Searching for resources

As part of the task assigned to him, Gabriele will search for available materials, useful for the course. He registers at the LearnWeb2.0 page and can now access the user management service of LearnWeb2.0 and edit his preferences. He provides login data for his favorite Web 2.0 tools to allow for automatic single sign on to these applications. Because the LearnWeb2.0 Agent is installed in his browser, Gabriele can identify important learning resources and group them together using the aggregation functionality, described in the section 3.4. He starts searching on the Internet and on his desktop for interesting resources. He gathers MS Access Training PowerPoint documents through Slideshare, video tutorials on Microsoft Access through YouTube, links and bookmarks through his del.icio.us page using the filter masks, described in the section 3.1. He uses LearnWeb2.0 resource upload functionality in order to upload new resources into a corresponding Web 2.0 tool and add them to the

---

1 http://www.ecdl.it/ ECDL (European Computer Driving Licence)

2 http://www.aicanet.it/ AICA (Associazione Italiana per il Calcolo Automatico)

3 http://www.slideshare.net/caddo1975/ms-access-training-486265 Sheyahshe, M., Skaggs, M. © 2008

4 http://uk.youtube.com/watch?v=jED1vmgN70
LearnWeb2.0 platform at one stroke as described in the section 3.5. The resources can be aggregated together into a group. Such group itself can be added to LearnWeb2.0 as a resource.

As soon as Gabriele drags a resource or file (e.g. a podcast describing MS Office features) into the LearnWeb2.0 window, the Agent offers a selection of groups to which the resource or file can be added. Gabriele collects materials and bookmarks, using filter masks and annotation functionality described in the section 3.1. All resources can be accessed and shared on a collaborative basis by every LearnWeb2.0 user and, in particular, the other team members. Gabriele can now create a new Advanced Access group and can drag the podcast into it.

2.2 Sharing research results
In the meantime Daniela looks for evaluation resources (e.g. Tests and quizzes) and uses LearnWeb2.0 in a similar way. While looking for exercises, she also finds other useful materials on the web. Because LearnWeb2.0 is a collaborative environment, Daniela can easily check to see whether Gabriele already found similar resources based on the query tags, as described in section 3.1. If not, she can decide to drop a short message into Gabriele’s browser using SpreadCrumbs, described in section 3.3. Daniela asks him to assess the item and to decide if it is worthwhile including it in the Advanced Access group.

Daniela browses through the search results and finds a group of related resources created by a colleague from a partner university who has already solved a similar task, and adds them to her Assessment group. She comments, tags and rates the resources. Finally, she shares the Assessment group with other ICT technicians in her university.

When a team member adds a new resource, the LearnWeb2.0 Agent provides notification to all work group participants. Gabriele and Daniela both participate in the Advanced Access and Assessment groups. Daniela gets notified when Gabriele adds new contents and can search for exercises in order to provide appropriate assessment related to the topic indicated by the notification. Other ICT technicians, including those from other Universities, are interested in sharing resources and in contributing to the group. For example, Daniela can easily share all resources tagged with “Photoshop, filters” to all her friends that are working on a specific project which requires the use of a graphics program.

In this way, they can benefit from notifications arrived from a colleague at the University of Pavia, or from a partner university, who have already solved a similar task. For example, the University of Pavia has an annex in Cremona where the Faculty of Musicology is located. Students in Cremona also want to use the online course envisaged, so the technicians in Cremona are interested in providing resources as well. Luigi has previous experience in creating online courses. He accesses the LearnWeb2.0 platform as an external guest, monitors the progress of the Pavia group’s work and can give them additional advice.

2.3 Aggregation and sequencing
Ivana is responsible for organizing the contents to be followed in the ECDL/ICDL Syllabus. She bases this on the work done by the other colleagues, creating a Syllabus group and organizing contents according to the specific requirements of the learning course. She can access the LearnWeb2.0 home page to get an overview over the various resource groups created so far. She can also add own related resources. Ivana can browse through the complete set of search results obtained so far, as well as look up the details of a particular resource. She can also comment, tag and rate resources using the annotation functionality, described in section 3.2. In the groups created by Gabriele and Daniela, she will find related resources which may be included in a learning module. For example, she is now in a position to include multimodal resources (e.g. animations) in a learning module, possibly as a multimedia section in the course. Ivana

---

5 http://whitpapers.zdnet.com/abstract.aspx?docid=178298
7 http://www.docnmail.com/tests/computers/database/access.htm
8 http://aicnet.net/certificazioni/ecdl/advanced-level/syllabus,
aggregates the contents into a sequence using the Learning Design Editor, described in section 3.4.

3. LearnWeb2.0 Functionalities
Knowledge resources related to specific user activities can be aggregated into groups (e.g., learning courses, events). In the context of lifelong competence development these groups are never complete, as knowledge resources appear, change and get outdated dynamically. LearnWeb2.0 supports aggregation of knowledge resources as well as tagging and classification on the resource and group level. So far, however, all these activities have to be applied manually to every single resource, consuming valuable user time.

Motivated by our scenario, in the next LearnWeb2.0 version we want to enable the user to specify the properties underlying the specific aggregation, classification, tagging or sharing decision, such that the system can perform event-driven group updates, e.g., automatically classify and share newly inserted knowledge resources.

In this section we describe the new functionalities corresponding to the requirements in our scenario described in the previous section, which will be available in the next LearnWeb2.0 platform release, after appropriate discussion with all TENCompetence partners concerned.

3.1 Resource selection and sharing
A LearnWeb2.0 user can select and share a set of resources based on a common property in order to support the search task of the Pavia team, as described on the section 2.1. Such property can be a tag, file type, timestamp and other properties or their combinations – a filter mask. A frequently used query mask can be stored as a standing query [5] in the user profile, enabling quick access to an up-to-date set of resources sharing the specified property as described in the section 2.2.

One further application of a standing query is to automatically share a suitable resource as soon as it is added to the repository and annotated. LearnWeb2.0 supports collaborative search with automatic resource annotation. Once a resource in the search result list is selected, it is automatically tagged with the corresponding query terms. These tags can later on be used for resource recommendations to friends and colleagues.

The search function of LearnWeb2.0 is not limited to a single Web 2.0 application, but integrates search results from a number of supported Web 2.0 sources. In the next release, LearnWeb2.0 will provide user-specific search functionality to enable a homogenous integrated view on all knowledge resources stored at the various Web 2.0 accounts of the particular user. This way the user will be able to place search queries limited to her distributed Web 2.0 virtual working space analogously to desktop search on a local machine.

3.2 Annotating search results
Manually adding new resources to LearnWeb2.0 or updating metadata of existing knowledge resources is time consuming. Therefore the next release of LearnWeb2.0 will provide selection and annotation functionality for a set of selected resources using a filter mask as described in the scenario section 2.1.

After selecting a set of knowledge resources, the user is forwarded to the editing page where she can perform metadata update on all selected resources at once. For example, the user can add new tags to the whole resource group, or assign all selected resources to a specific category as well as specify the time, location, language and access rights.

3.3 Bookmark sharing and notification
During web browsing the user might identify and annotate interesting webpages and notify partners and friends about these pages, as described in section 2.2. LearnWeb2.0 will support this functionality using the tool SpreadCrumbs [4]. SpreadCrumbs enables users to create a sticky note on any web page, specifically addressed to one or several friends of the user. This note appears each time the recipient, a friend or a colleague of the user, enters the web page. Additionally, the user finds all notifications addressed to him in her LearnWeb2.0 profile.
3.4 Resource aggregation and sequencing
The user may also be interested in aggregating a set of knowledge resources related to the same learning activity or competence as described in the content aggregation and sequencing scenario in section 2.3. LearnWeb2.0 uses the Web 2.0 tool GroupMe! for resource aggregation. Currently only manual resource grouping is supported. In the next release we want to support grouping based on a filter mask.

Furthermore a standing query will enable automatic instant updates of the group with new suitable resources as soon as they are added to the LearnWeb2.0 system.

LearnWeb2.0 provides web services which export its functionality to third party applications. This allows us to provide the LearnWeb2.0 services for any sequencing tool. In the TENCompetence environment resources are organized using the Learning Design Editor ReCourse [2].

3.5 Resource upload
LearnWeb2.0 is an integrated environment which supports upload of resources using a common user interface and storage of these resources in the distributed Web 2.0 virtual working space of the user. The Web is not the only source where the user can find resources related to a competence. Some useful resources might be located on the user’s desktop or other devices like a camera. LearnWeb2.0 provides a possibility to directly upload the resource from the desktop or an external source to a suitable Web 2.0 tool and to annotate it as described in scenario section 2.1.

In order to upload a resource, the user selects a file to be uploaded and the Web 2.0 target tool. LearnWeb2.0 supports upload to Web 2.0 tools through the tool API. Upon upload, the user specifies an access policy to the resource (private or public) as well as further metadata and uploads the resource which afterwards appears in her user profile.

4. Conclusions
In this paper we used a real-world scenario to identify and discuss further challenges for integrating social software tools into our LearnWeb2.0 infrastructure. We sketched the main required activities and described the most important functionalities, to be provided in the next LearnWeb2.0 release.

Moving from manual resource aggregation and annotation, we will provide resource aggregation based on an automatic filter mask, and standing queries will enable instant updates of new resources.

In general, the first release of LearnWeb2.0 has focused on interoperability. The functionalities in the next release will focus on community support and sharing in a Web 2.0 environment, moving the LearnWeb2.0 environment more towards social network software. Examples include automatic notification along a friendship graph as well as further promotion of the internal communication among LearnWeb2.0 users.

5. References