Towards Community-Driven Development of Educational Materials: The Edukalibre Approach

Jesus M. Gonzalez-Barahona[1], Vania Dimitrova[2], Diego Chaparro[1], Chris Tebb[2], Teofilo Romera[1], Luis Canas[1], and Julika Matravers[2], and Styliani Kleanthous[2]

[1] Grupo de Sistemas y Comunicaciones, Universidad Rey Juan Carlos, Mostoles, Madrid, Spain
[2] School of Computing, University of Leeds, Leeds, UK
{jgb, dchaparro, teo, lcanas}@gsyc.escet.urjc.es,
{vania, chrispy, julika, stellak}@comp.leeds.ac.uk

Abstract. The libre software development model has shown how combining collective intelligence can lead to revolutionary methods that underpin major software advancements. This paper argues that the time is ripe to examine the application of this model to the development of educational materials where not only teachers but also students can become actively involved in the creation of educational content. The paper describes a novel, *truly open* platform, developed within the Edukalibre project [1], to support the creation of collaboratively constructed educational materials. The paper presents the Edukalibre collaborative editing system that provides easy access to core technologies composed of a document repository with version control management and conversion tools to produce several formats for each document. Two different collaborative editing interfaces - COLLAB and ConDOR - have been implemented. Based on evaluation studies with COLLAB and ConDOR, we discuss several pedagogical and technological issues related to the deployment of community-driven development of educational content.

1 Introduction

The libre [2] software development method has tremendously changed the way software is being produced and deployed [1,2] in a number of domains, ranging

[1] The Edukalibre project was funded by the European Commission's, Socrates/Minerva program, grant 110330-CP-1-2003-1ES-MINERVA-M. This paper reflects the views of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

[2] We use the term "libre software" to refer both to "free software" and "open source software". The term "libre" lacks the ambiguity of "free" (which means both "gratis" and "free as in freedom"), and makes reference to the liberty of change that the user of the software gains with it. In short, libre (free, open source) software is defined by the freedom of use, study, modification and distribution that is granted to those receiving a copy of the programs.

W. Nejdl and K. Tochtermann (Eds.): EC-TEL 2006, LNCS 4227, pp. 125–139, 2006.

from operating systems (Debian, FreeBSD, Fedora) to desktop environments (GNOME, KDE), web browsers (Mozilla, Firefox), web servers (Apache) or office suites (OpenOffice.org) [3,4,5,6]. It is now widely recognised that the libre software development model, where software is produced by communities of practitioners sharing experiences, methods, and code, has led to revolutionary methods that produced major advancements in software development [7,8].

This paper argues that the time is ripe to deploy the libre, community-driven development model to the creation of educational content. Following the impact on software development, it can be expected that the community-driven development model will lead to innovative educational paradigms and will have a great impact on the way the web is used for teaching and learning. There is a growing interest among the educational community to adopt aspects of the open development model. The first step is being undertaken by the *open learning content* idea which has been taken on board with great enthusiasm in university education[9], and has been implemented in a number of projects, e.g Open Learning Support [10], MIT OpenCourseware [11], Open Learning Initiative [12], Connextions [13], CASCADE[14], WIKI [15].

To fully deploy the community-driven development model both educational practitioners and students should become actively involved in the creation and distribution of open educational content. Until recently, technical challenges made it very difficult to support truly open, dynamic, educational resources constructed collaboratively by large groups of teachers and even students. Nowadays, the libre software community has created a vast amount of technologies to support their practices. However, these technologies have not been explored fully in an educational context. Moreover, many of the existing technologies for collaborative development of software suffer from two problems: (1) they are suitable for software developers but are not intuitive enough to be adopted by teachers and students and (2) they normally tackle small tasks, which is convenient for developers who usually use several tools, but can be inappropriate for use by wide and diverse teacher and learner communities.

New architectures are needed to effectively support the collaborative construction of open educational resources [16]. Moreover, these architectures should themselves be open, enabling customisation and deployment in different settings. An example of such an architecture is proposed in this paper. We describe a novel, *truly open* platform to support the creation of free, collaboratively constructed educational content. The platform was developed within the Edukalibre project [3] which examined the connection between libre software development and the creation of open content for education.

[3] The project was funded by the European Commission under the Socrates/Minerva program and ran in the period October 2003 - December 2005. It was coordinated by University Rey Juan Carlos (Spain), and included as partners teams from the University of Leeds (United Kingdom), University of Porto (Portugal), University of Karlsruhe (Germany), University of Lugano (Switzerland), and the Academy of Sciences of the Czech Republic. Project web site: http://www.edukalibre.org.

The paper will first discuss key characteristics of the community-driven development model and its link to the production of educational materials (Section 2). Section 3 will then present the architecture of the Edukalibre system. Two tools that provide a user interface to the system - COLLAB and ConDOR - will be described in Section 4. Section 5 will discuss the deployment of Edukalibre in university teaching. Finally, in the conclusions, we will discuss the contribution of Edukalibre to technology-enhanced learning and will outline future work.

2 Community-Driven Development and Education

The libre software community is one of the best examples of an online community within which there are smaller communities of practice [17] of people solving problems that arise within a particular project and producing solutions that can be used and modified by others [18,19]. The Edukalibre project explores how this community-driven development model can be applied to the education camp to facilitate collaborative development of educational materials. Therefore, we will first outline the key characteristics of the libre software model.

There are several specific issues found across most libre software projects, as discussed in [7,20,3]:

- Frequent and early release of software to enable opportunities to gain external feedback and contributions.
- Decentralised quality maintained by many individuals, including those outside the group of developers.
- Libre software projects, especially when of a certain size, are carried out by a number of geographically distributed developers. They seldom (if ever) see each other face-to-face, and work in a coordinated fashion without formal hierarchies.
- Development is done asynchronously by using software tools for coordination (e.g. mailing lists, bug tracking systems, version control systems, software repositories) and hosting facilities (e.g. SourceForge[4] or Savannah[5]).
- There is a mixture of voluntary and paid work.

In comparison, the deployment of a community-driven, libre development model to education may address the following issues:

Collaboration by educators: Materials will be produced by groups of educators, usually in different institutions, and geographically dispersed. Since the curricula are similar in many cases, it seems reasonable that different teachers have similar needs, and can collaborate in the making of their materials. The collaboration has the implication of sharing the work, coordination, and consensus. Similar situations are found in libre software projects, and it is expected that similar results can be obtained. In the current European context, where collaboration among universities of different countries is encouraged at many levels, this will

[4] `http://sourceforge.net`
[5] `http://savannah.gnu.org`

increasingly become a common situation, elements of which are demonstrated in several successful European projects, e.g. ARIADNE [6] and EducaNext [7].

Active contribution from students: Materials will also be used, commented, and possibly modified by students. In fact, in many cases, students are already producing their own materials, based on those provided by educators. In the same way that users can contribute to libre software projects highlighting bugs or even proposing useful modifications to the programs (if they have enough knowledge), students (users of the materials) can contribute to the continuous improving of quality. From this point of view, frequent and early release could be as important as it is in the case of libre software.

Intuitive tools: Educators and students will need simple to use, yet powerful, tools to be able to collaborate in the way libre software developers do. For instance, it will be important to mimic the functionality of version control systems and compilation systems, specifically oriented to the production of educational materials. At the same time, authors should be able to use tools in such a way that coordination and work in common is becomes an intuitive process.

Public availability: The public availability of produced materials will enable the collaboration of third parties, such as other professors or students from other institutions who may find the materials useful, and can contribute to their further development.

Agreed licensing terms: Several licensing terms may be explored, some of them allowing for publishers to take the materials and distribute them for profit, in the same way that there are companies distributing GNU/Linux based systems composed only of libre software. The current interest created by some initiatives related to open content (such as Creative Commons[8], the Open Archives Initiative[9], and the the Public Library of Science[10]) is caused by many other projects exploring, particularly in the scientific context, the advantages of this approach.

Of course, this model is difficult to explore without the proper tools and some user groups ready to test and try it, which was the main focus of the Edukalibre project. The following sections will present the Edukalibre system and will discuss case studies in university teaching environments.

3 The Edukalibre Collaborative Editing System

The Edukalibre system is composed of several software components, some of which are widely used in current libre software projects and others have been developed specifically within the project. The architecture of the system is intended to be simple, yet flexible, reusing as many already available components as possible, so that the project can focus on its goals whilst incorporating

[6] www.ariadne-eu.org
[7] www.educanext.org/ubp
[8] `http://creativecommons.org`
[9] `http://www.openarchives.org/`
[10] `http://plos.org`

already tested and widespread components, such as the Subversion[11] version control system or the Moodle[12] learning management system. The use of standards compliant technologies and ideology of open access to the different components of the system has been a core design principle. The Edukalibre system has been developed to be as open as possible, using libre software technologies throughout and allowing access to content in as many different ways as is practical.

This paper presents the collaborative editing part of the Edukalibre system. The complete Edukalibre suite of tools includes also a graphical interactive students monitoring tool and a Wiki compiler for mathematical formulas [13]. Collaborative editing with the Edukalibre system enables users to choose from a wide set of tools (from easy-to-use word processors, such as OpenOffice.org[14], to less common XML-based editors), and includes version control facilities and automatic conversion to many end-user formats (ranging from PDF, ready to be printed on paper, to decorated HTML, suitable for previewing when creating a new version or for reading online).

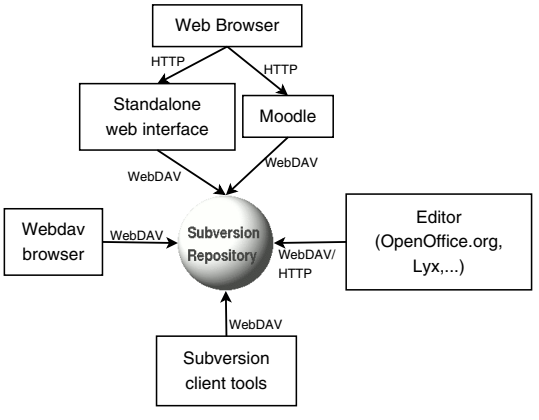

Fig. 1. Components of the collaborative editing subsystem

Figure 1 shows the architecture of the collaborative editing subsystem. The core is a *document repository*, built on top of a *version controlled storage system*, with document conversion and management tools written in `Python`[15] [21]. The main components of the Edukalibre collaborative editing system are:

Version control: Every document created in the system has a main version and a history of older revisions or branches. Each revision can be accessed or modified at any time, and any old revision can be set up as the main version of the document, or assigned as a new branch (for example to adapt higher education teaching materials for a secondary education audience). Currently, the platform uses Subversion as the control version system. Subversion is in some sense an evolution of the more traditional CVS[16], which fixes many of its problems, and adds new features.

[11] `http://subversion.tigris.org/`
[12] `http://moodle.org`
[13] See www.edukalibre.org for a complete description of all Edukalibre tools
[14] `http://openoffice.org`
[15] `http://python.org`
[16] `http://www.cvshome.org`

Editors and word processors: Our goal is not to create new editors, but to provide ways in which existing editors and word processors can interact in the most flexible way with the system. Authors already have their preferred tools, so our goal is to support as many authoring tools as possible, and offer the same version control and document conversion features to all authors, irrespective of their preferred editing application. The current Edukalibre platform allows a document to be opened, modified and stored using OpenOffice [17] which can edit, in addition to files in its own format, DocBook/XML[18] files (with the help of some code which the project has improved) and Microsoft Word documents. However, any other word processor can be used to modify the documents of the system, if it can manage DocBook/XML or LaTeX[19], the currently supported document formats. Since both are plain-text formats, any text editor can also be used, provided the user knows the syntax of the format. A simple web editor is also provided.

Tools for format conversion: Converting documents from the base formats to other formats more suitable for reading (such as HTML or PDF), is enabled via a suite of conversion tools. Most of the conversions from XML formats are performed using XSLT[20] stylesheets. The XSLT processors used are xsltproc and xalan[21]. Conversions to PDF and Postscript are done using TeX/LaTeX and Ghostscript[22]. A wiki-like system supporting conversion from wiki format to DocBook/XML and vice-versa has been developed.

Web interfaces: One of the goals of the project is to allow for multiple interfaces to the system, and to let users choose their preferred writing tools. In addition to accessing the system directly with a text editor plus a Subversion client (supporting command line editing with unix text editors and familiar to academics mainly in the IT field), other interfaces already provided are:

- A collaborative editing system (COLLAB), which is a Python-written web interface that permits authors to perform common actions in the system, such as listing of documents or visualization of the history of a document.
- A PHP-based groupware application (ConDOR), which is a web based file system view for the repository and provides multiple repository support together with discussion and messaging tools.

Communication protocols: Standard protocols are used to connect the different components. They include the ubiquitous HTTP [22], and WebDAV [23] (an extension to HTTP tailored to the needs of collaborative editing and management of files on remote web servers).

From the perspective of an author collaborating to write some material, the platform provides the two interfaces - COLLAB and ConDOR. Currently, both

[17] `http://www.openoffice.org/`
[18] `http://www.docbook.org/`
[19] `http://www.latex-project.org/`
[20] `http://www.w3.org/TR/xslt`
[21] `http://xml.apache.org/xalan-j/`
[22] `http://www.cs.wisc.edu/~ghost/`

interfaces are provided in two forms - as standalone web applications and as integrated components in Moodle. At the time Edukalibre evaluation studies were conducted, see next section, COLLAB was a standalone web interface, while ConDOR was integrated in Moodle. This enabled the evaluation of two types of access and tested the flexibility of the Edukalibre platform.

From the perspective of a developer wishing to incorporate access to an Edukalibre repository into their applications, the platform provides a simple PHP[23] API and Subversion/WebDAV access to the repository for other development platforms.

4 Collaborative Editing Interfaces

In the following subsections, the two collaborative editing interface applications implemented within Edukalibre are described. They show both how the functionality of the system can be integrated with external tools, and how the architecture is modular enough to make room for many different ways of interaction.

4.1 COLLAB: A Python-Driven Web Interface

This is one of the possible interfaces to the system, and allows the users to perform some of the common tasks using just a web browser. A screenshot of the COLLAB interface can be seen in Figure 2. The main page of the interface shows a list of documents in the system, with some information about each of the documents. The interface provides a way to create new documents. An RDF channel with information about the documents recently modified or created is also provided. The information for each document includes:

- Listing of its version history, with the release date and the comments from the release author.
- Listing of links to all end-user formats for every version, so that users can get any format of any version of any document.
- Editing and downloading options for each version, including the possibility to download the base format, modify it and upload the new version using the web interface, or to edit it on-line with a simple web editor.
- Forms to update the document, uploading a new version (written with a standalone editor such as OpenOffice.org).
- Decorated HTML version of the latest version of the document to show a quick preview.

4.2 ConDOR: An Intuitive PHP-Driven Groupware Application

One of the core deliverables of the project was to create a *user friendly* working environment which would allow us to evaluate the effectiveness of web-based Construction of Dynamic Open Resources (ConDOR), see Figure 3.

[23] `http://php.net`

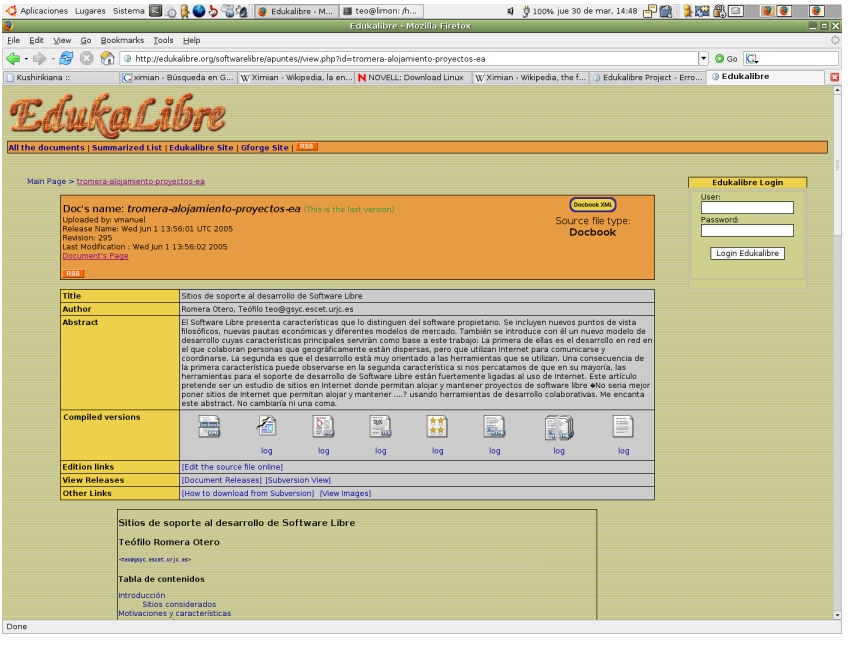

Fig. 2. COLLAB interface: example from evaluation study 3, described in Section 5

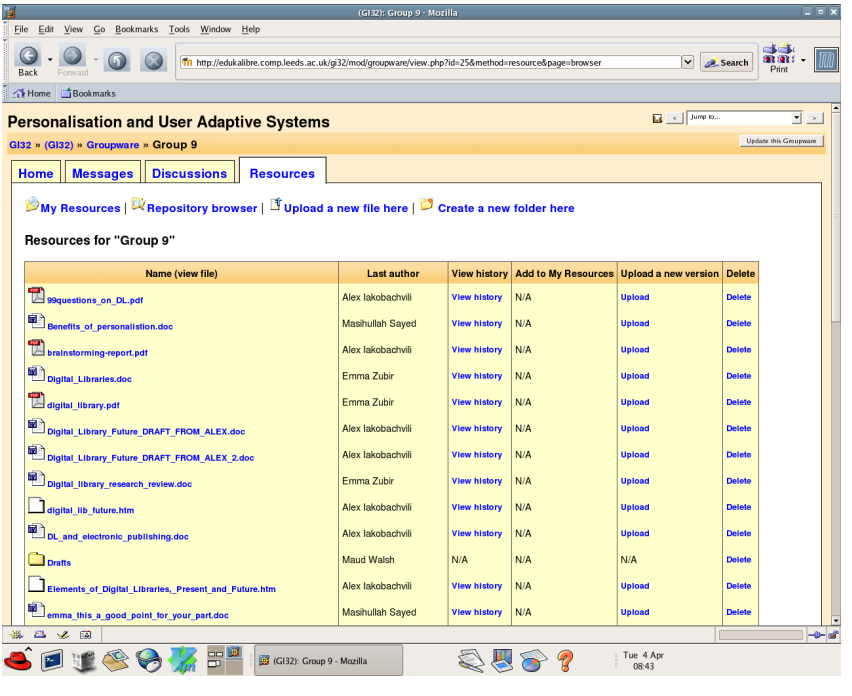

Fig. 3. ConDOR interface: example from evaluation study 2, described in Section 5

ConDOR is a bespoke groupware application which can run standalone, or as a modular component of the libre software learning management environment Moodle. The requirements specification for the groupware was taken from a questionnaire that was filled in by all partners of the project, enquiring into the scenarios in which the platform would be tested. ConDOR provides "explorer style" access to the document repository, allowing intuitive navigation of the file and folder structures, and simple tools for uploading files and creating new folders. To prevent excessive repeated navigation of large folder structures, a "My Resources" section is provided within the ConDOR resource area to allow authors to bookmark documents of special interest within the resource tree. From either the main resources browser or the "My Resources" area, authors can download a document in any supported format (currently XHTML, PDF, docbook and plain text), upload a new version of the document, browse the history of the document and add/remove documents from their "My Resources" list.

The repository allows authors to add any type of file, and put it under version control. Extra features such as format conversion are available for OpenOffice and LaTeX documents. In addition, trial users have been using the repository to manage different kinds of files. A graphic designer using the system used it to manage multiple versions of an Adobe Photoshop document he was working on with another designer, and a systems administrator has used the system to manage various versions of Windows configuration files.

5 Evaluation Studies

The evaluation of the Edukalibre system, given its libre nature, had to deal with several challenges. On the one hand, the highly iterative development process required frequent releases of software (sometimes not fully tested) and demanded quick feedback. On the other hand, user studies in real settings were required to examine the feasibility of the deployment of the libre development model in education. To address these challenges, a flexible, development-led evaluation approach was followed which included two phases - inspection by inspectors and several small scale studies with different Edukalibre tools. The Edukalibre evaluation is presented in detail elsewhere, see [24]. We will refer here only to user studies with COLLAB and ConDOR which highlighted important technological and pedagogical issues of the community-driven development of educational materials. The following user studies were conducted with the Edukalibre collaborative editing tools:

Study 1: Students preparing a group report on an environmental case using ConDOR. The study was conducted at the University of Karlsruhe with 14 students attending an advanced Industrial Engineering seminar [25]. Working in separate groups, the students spent four months researching environmental cases concerning fictitious companies. Each group produced a collaboratively written report based on materials collected by the members. The seminar was run with Moodle and the students used ConDOR for sharing re-

sources and for collaborative writing. The process was monitored by 2 teachers via the Edukalibre student monitoring tool [26]. The students' feedback was collected with a questionnaire at the end of the seminar.

Study 2: Students preparing learning resources on a new topic using ConDOR. The study was conducted at the University of Leeds with 55 students attending a final year undergraduate course on Personalisation and User-adaptive Systems. The students were split into 13 groups. Each group had to research a topic new to them (e.g. personalisation for e-health, digital libraries, or e-learning) and prepare a report which, together with the supporting electronic resources collected, was later used as a learning material by the whole class. The course was run with Moodle and the students used ConDOR for the collaborative writing activities which spanned in one month. At the end, a questionnaire was conducted to collect the students' feedback.

Study 3: Students writing and commenting on papers using COLLAB. The study was conducted at the University Rey Juan Carlos with 24 PhD students who attended a three month course on libre software. The students had to prepare a paper on a topic relevant to the course. They had to help each other by editing and commenting on at least two of the papers being prepared by their peers. The students wrote their papers during the duration of the course and, at the same time, they could read and review papers by their peers. They used COLLAB to download, review and correct papers of their peers and to create and maintain their own papers. At the end of the course, a questionnaire was carried out to find out the users' opinions of the tools. Many of the participants were highly trained programmers, IT technicians or computer engineers, so they offered valuable feedback to the development team and also had less problems to use the tools, since they were familiar with concepts like version control systems and collaborative editing.

Study 4: Teachers preparing a course using COLLAB. The study was conducted at the Academy of Sciences of the Czech Republic. 4 teachers worked together on the preparation of a course on Bayesian Networks suitable for research students. The teachers undertook several iterations of the learning materials. The course included textual descriptions and interactive examples. The teachers gave informal feedback about the usefulness of the tool.

Due to space constraint, we cannot present full details of each study, the reader is directed to the corresponding papers [24,25] and project deliverables [24]. The following section will focus on lessons learned about the deployment of the libre development model to education, based on the combined results from the studies.

6 Using COLLAB and ConDOR for Community-Driven Development of Educational Materials

This section will focus on what the studies with COLLAB and ConDOR told us about the process of community-driven development of educational materials. The results can broadly be divided into pedagogical and technical aspects.

[24] Available from the project web site www.edukalibre.org

6.1 Pedagogical Aspects

Team work and collaborative writing. All studies exhibited the three stages of team work: *forming, functioning, finishing* [27]. Since the group members were familiar with each other, we expected that the forming would be quick and there would be a fair degree of trust among group members. However, most student groups had a slow start and the questionnaire results indicated trust problems. For example, 50% of the students in studies 1 and 2 were reluctant for others to be able to see what they write and 64% would not like their writings to be changed by others. In contrast, most participants (83%) in study 3 and all participants in study 4 were willing to share their incomplete writing so they could benefit from reviews, comments and corrections of their peers. Analysis of the discussions with ConDOR in study 2 showed that most groups spent considerable time clarifying the members' expertise and preferences and agreed on what section/subtopic each person was responsible for. The functioning phase showed that most participants actively used the collaborative editing tools. For example, 46% of the students using ConDOR felt that the tool helped them to be more efficient. Most concerns were with regard to the slow access to the document repository, yet 79% were positive that they could share resources with others. In study 2, several conflicts with regard to members' contribution arose and a spilt of three groups was inevitable. Some of the materials produced by the students were not fully coherent and included repetitive or contradicting opinions. The groups who were successful (judged by the quality of their reports) had one or two cognitively central members [27] who would take the leading role in decisions about structure, topics, or resources. In each study, all materials produced were made public to the other participants in the study. While in studies 1 and 2, other groups could only comment on the quality of the produced materials, in studies 3 and 4, everybody could not only comment but also alter what was produced by others. Students in studies 1 and 2 appreciated the feedback from their peers but required some kind of central comment from teachers. This indicates the need of some quality control that is not required in the libre development model (see Section 2) but is important in educational settings.

Building communities of practice. Clearly, there was learning benefit from all studies involving students. Students did not only practise researching a topic and writing materials to be used by others, but also became exposed to a wider set of opinions, engaged in discussions clarifying domain aspects, and became active participants in the course. The evaluation studies showed that the pedagogical impact from engaging students or educational practitioners in collaborative creation of educational materials would be associated with the forming of communities of practice where students and educational practitioners, often from the same or related background, are coming together to share ideas and experience and to tackle common problems and issues [28]. For example, the students in Study 3 spontaneously helped each other with comments, corrections and contributing sections to papers of their peers. Almost all of the students commented that the global quality of the papers they wrote grew because of the help they offered each other. Study 2 also showed elements of learning in

communities: initial analysis of the group interactions indicates evidence that by sharing resources and ideas and engaging in a joint task the groups acquired knowledge and the members improved their understanding of the domain.

Integration within the learning environment. The Edukalibre experience showed that engaging students in the collaborative construction of educational materials would be effective only if it is integrated within the overall learning environment. Teachers involved in studies 1, 2, and 3 had to carefully design the collaborative writing exercises linking them to course learning objectives. This link was made clear to the students prior to each study. In studies 1 and 2, which were conducted with undergraduate students, the collaboratively produced reports contributed a small percentage to the students' course assessment. In Study 3, conducted with research students (most at PhD level), all participants were familiar with the libre software development model and its benefits. For them, the study was a valuable experience to examine the feasibility of transferring this model to educational settings.

6.2 Technological Aspects

Flexibility. One of the main design principles of Edukalibre was to provide flexible user access. The collaborative editing tools enabled the conversion between several formats which was appreciated by the participants in studies 3 and 4 who had experience with (study 3) and understanding of (study 4) the benefits of the libre development model. In contrast, the students in studies 1 and 2 did not use the format conversion tools despite being reminded by the teachers several times. Also, the students in these studies did not take the full advantage of the version control functions, whilst this was acknowledged as a very useful feature by the participants in studies 3 and 4. The participants in studies 1 and 2 stressed the need to have an editor integrated within the system, as they found it inconvenient to use an external editor (most of them used Microsoft Word and few used Open Office) which slowed the download/upload/commenting process. An html/docbook editor has been included in the last version of ConDOR released at the end of the project.

Holistic design. The studies clearly pointed out the need for holistic design. It became apparent that resource sharing, communication, and collaborative writing would happen together and should be supported within one environment (as opposed to fragmented tools used in libre software development). This was partly taken into account in the design of ConDOR, but the studies revealed that more features had to be integrated in the interface applications, e.g. appropriate editing tools, links between discussions and documents, and association of roles to participants. Since Edukalibre was designed as a truly open platform, it can easily be extended with new functionality, as discussed above.

7 Conclusions and Further Work

The Edukalibre project makes two important contributions to technology-enhanced learning, namely:

Deployment of the libre development model in education: Edukalibre explores the field of collaborative production of educational materials from a novel point of view: can the common practices of the libre software community be applied to the development of educational content. The project was just an initial step which focused on the development of appropriate tools to support community-driven creation of educational content, and the deployment of these tools in small studies in realistic educational settings. Evaluation studies with COLLAB and ConDOR confirmed that it would be feasible to deploy a more open, community-driven model for the creation of educational resources. Distinctively from other projects that explore sharing, and even collaborative creation of educational resources mainly by teachers, Edukalibre examined opportunities for active engagement of students. Our experience showed that there was a great potential for creative scenarios where students could engage in collaborative writing experiences to create resources that would be used by other students or could help their peers with writing papers.

Deployment of Web 2.0 to the development of technologies for education: The second major contribution of Edukalibre is to demonstrate the application of Web 2.0 to the development of technologies for education. The project followed key tenets of Web 2.0 [29], namely, harnessing collective intelligence, lightweight programming models, sharing and reuse of existing components and tools, build applications that enrich the users' experiences offer creative ways to quickly build effective systems. The project has developed (mainly by reusing already existing components) a web-based system which is built around a Subversion-based repository for document sources, together with access methods to it and automated generation of final formats. The Edukalibre architecture is open and modular in nature, and makes it simple to add new functionality by connecting new subsystems. In some sense, Edukalibre implements a wiki-like syntax, with simple to use interfaces integrated in environments (such as office suites or learning management systems) common to educators. All in all, it lowers the technical barriers for educators to explore the libre software development model applied to educational documents. Also, the whole system is composed of libre software, which makes it easy to try, install and adapt to any environment.

Being an initial step, the project did not examine the full potential of the deployment of the libre development model. Notably, all studies included participants who knew each other and could meet face-to-face. Further studies are required to examine how the Edukalibre system can be tailored and extended to enable the construction of educational resources by large, geographically remote communities. In all scenarios explored so far, we dealt with university settings (which was the focus of Edukalibre). There is notable interest in methods for community-driven creation of content outside the university communities, for example schools [30] and digital libraries [31], and studies in such settings should be conducted. Finally, Edukalibre focused on the creation of educational materials but did not explore how these materials would be reused by different teachers and students. For this, significant extension of the Edukalibre editing tools would be required to generate appropriate meta-data and to export the created resources

as learning objects [32]. Finally, some participants pointed that the collaborative editing tools should be customisable, so that the communities could decide what tools would suit them best. Furthermore, offering support tailored to the particular community (e.g. pointing at missing topics, inactive participants, or dominant members) will be beneficial.

Acknowledgements. The authors are grateful to all partners from the Edukalibre consortium for the stimulating discussions that helped clarify the ideas behind the collaborative editing system, and especially to the teams from Lugano, Karlsruhe and Prague who took active part in planning and conducting the evaluation studies. Edukalibre was funded by the European Commission.

References

1. Bezroukov, N.: Open source software development as a special type of academic research. First Monday **4** (1999)
2. Hassan, A.E., Godfrey, M.W., Holt, R.C.: Software engineering research in the bazaar. In: Proceedings of the 2nd Workshop on Open Source Software Engineering at the 24th International Conference on Software Engineering. (2001)
3. German, D.M.: The GNOME project: a case study of open source, global software development. Software Process Improvement and Practice (2003) 201–215
4. Gonzlez-Barahona, J.M., Lpez, L., Robles, G.: Community structure of modules in the apache project. In: Proceedings of the 4th Workshop on Open Source Software Engineering. 26th International Conference on Software Engineering, Edinburgh, Scotland, UK (2004)
5. Reis, C.R., de Mattos Fortes, R.P.: An overview of the software engineering process and tools in the Mozilla project. In: Workshop on Open Source Software Development. (2002)
6. Lameter, C.: Debian gnu/linux: The past, the present and the future. In: Free Software Symposium 2002. (2002)
7. Raymond, E.S.: The cathedral and the bazar. First Monday **4** (1997)
8. Koch, S., ed.: Free/Open Source Software Development. Idea Group, Inc. (2004)
9. Siemens, G.: Open source content in education: Developing, sharing, expanding resources. Pitch: peer reviewed online journal in Instructional and Learning Technology (2003)
10. Utah University Open Learning Support `http://ols.usu.edu/`.
11. MIT OpenCourseWare `http://ocw.mit.edu/index.html`.
12. Carnegie Mellon Open Learning Initiative `http://www.cmu.edu/oli/`.
13. Connextions: Sharing knowledge and building communities `http://cnx.rice.edu/`.
14. Sapsomboon, B., Andriati, R., Roberts, L., Spring, M.B.: Software to aid collaboration: Focus on collaborative authoring. Technical report, School of Information Sciences, University of Pittsburgh (1997)
15. Wikipedia `http://www.wikipedia.org/`.
16. Noel, S., Robert, J.M.: Empirical study on collaborative writing: What do co-authors do, use, and like? Computer Supported Cooperative Work: The Journal of Collaborative Computing **13** (2004) 63–89

17. Wenger, E.: Communities of Practice: Learning, Meaning, and Identity. Cambridge University Press (2001)
18. Feller, J.: Thoughts on studying opens source software communities. In Russo, N.L., Fitzgerald, B., Degross, J.I., eds.: Realigning Research and Practice in Information Systems Development: The Social and Organizational Perspective, Boston, Kluwer (2001)
19. Raymond, S.E.: The Cathedral and The Bazaar. O'Reilly (2001)
20. Ghosh, R.A., Robles, G., Glott, R.: Software source code survey (free/libre and open source software: Survey and study). Technical report, International Institute of Infonomics. University of Maastricht, The Netherlands (2002)
21. van Rossum, G., Fred L. Drake, J.: The Python Language Reference Manual. Network Theory Ltd (2003)
22. Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T.: Hypertext transfer protocol – HTTP/1.1, RFC 2616. Technical report, Network Working Group, IETF (1999)
23. Goland, Y., Whitehead, E., Faizi, A., Carter, S., Jensen, D.: HTTP extensions for distributed authoring – WEBDAV, RFC 2518. Technical report, Network Working Group, IETF (1999)
24. Botturi, L., Dimitrova, V., Matravers, J., Tebb, C., Withworth, D., Geldermann, J., Hubert, I.: Development-oriented elearning tool evaluation: the edukalibre approach. In: ED-MEDIA: World Conference on Educational Multimedia, Hypermedia and Telecommunications, Montreal, Canada. (2005)
25. Geldermann, J., Treitz, M., Rentz, O.: Teaching integrated technique assessment based on open-source software. In: EUNITE Conference: European Symposium on Intelligent Technologies, Hybrid Systems and their implementation on Smart Adaptive Systems, Aachen, Germany (2004)
26. Mazza, R.: Gismo: a graphical interactive student monitoring tool for course management systems. In: Technology Enhanced Learning Conference, Milan (2004)
27. Ilgen, D.R., Hollenbeck, J.R., Johnson, M., Jundt, D.: Teams in organizations: From input - process - output models to imoi models. Annual Review of Psychology **56** (2005) 517–543
28. Lewis, D., Allan, B.: Virtual Learning Communities: a Guide to Practitioners. Berkshire, Open University, McGraw Hill (2005)
29. O'Reilly, T.: What is web 2,0: Design patterns and business models of the next generation of software. (2005)
30. Vuorikari, R.: Why europe needs free and open source software and content in schools. INSIGHT (2004)
31. Cushman, R.: Open educational content for public digital libraries. Technical report, The William and Flora Hewlett Foundation (2002)
32. Duval, E.: A learning object manifesto. In: PROLEARN NoE Deliverable D4.2. (2004)