International Journal of Engineering Education Vol. 27, No. 3, pp. 656–669, 2011
Printed in Great Britain
0949-149X/91 $3.00+0.00

Infrastructure Supporting Collaborative Project Based Learning in Engineering Education*

TANIA VASILEVA, VASSILIY TCHOUMATCHENKO
Department of Electronics, Technical University of Sofia, Sofia, Bulgaria. E-mail: {tkv@tu-sofia.bg, vpt@tu-sofia.bg}

MINNA LAKKALA, KARI KOSONEN
Institute of Behavioural Sciences, University of Helsinki, Helsinki, Finland. E-mail: {minna.lakkala@helsinki.fi, kari.kosonen@helsinki.fi}

The paper presents a web-based virtual environment aimed to promote project based collaborative learning and continuous work around shared objects. It is designed to provide specific affordances for joint development of shared objects, organizing related tasks as well as user networks and interactions. The environment provides a coherent set of tools in order to initiate and manage collaborative processes, to create, work on and share documents collectively, and to support synchronous and asynchronous communication among participants. A co-design framework, basic features, architecture, as well as supporting tools and their implementation are considered. The results of current field trials with the tools are also briefly outlined.

Keywords: collaborative learning; project-based learning; distributed environment/web-based environment; engineering education

1. Introduction

Contemporary changes in the business life, responses of companies to these changes and the fast alterations of available technologies pose a number of challenges to present and future engineers, as well as to the educational institutions, including:

- Collaborative work in the multidisciplinary, multinational, geographically dispersed teams;
- Creative work, communication skills, transfer and improvement of engineering knowledge.

In the conventional teaching methods in engineering education, students try to follow the routine solution procedures and focus on the ability to repeat what has been taught to them without necessarily understanding it. Meanwhile, engineering companies require abilities of collaborative work, team and task management, concept synthesis, decision-making and technology-supported communication from the graduates. Engineering education should offer students more opportunities to face the real challenges of working practices that are typical in professional life, thus creating environments for learning, which would provide these job-related skills [1].

Together with changes in society, conceptual frameworks, practices in school and at work also have to be varied to facilitate the development of corresponding individual and cultural competencies.

To prepare students for the practices of design and engineering in the 21st century, collaborative design exercises and projects has become an important element of many study programs [2–3]. Also the Bologna Declaration [4] and the implementation of the European Higher Education Area until 2020 [5] are promoting the use of active learning methodologies such as collaborative learning and project based learning. In addition to basic design skills, such as the ability to transfer theoretical knowledge to a particular design problem, to tolerate ambiguity, maintain the big picture, handle uncertainty and make informed design decisions, collaborative design also requires participants to coordinate and orchestrate joint efforts and to communicate and integrate multiple perspectives effectively [6]. Students need to be provided with opportunities to articulate their ideas, to test these ideas through experimentation and discussion, and to consider the connections between them.

Traditionally higher education and workplace learning have been aimed at making students acquainted with the up to date body of domain-specific knowledge as well as state-of-the-art practices, but today's students and employees are, increasingly, asked to make active use of and contribute to the creation of new knowledge. Towards this end, there is a need for new methods but also technological tools that support both students as well as practitioners in sustained efforts of innovation, knowledge creation and social participation around shared objects of activity.

1.1 Collaborative project based learning

Project Based Learning (PBL) has been established in the field of engineering as a promising method,

* Accepted 4 February 2011.

which promotes teamwork and communication skills as well as the management of complexities of professional practice [7]. Project Based Learning is a student-centred strategy that fosters active participation and engages students on authentic open-ended assignments simulating real-life challenges. One benefit of PBL is that it provides motivation for students to engage in learning that is self-directed and based on their own interests.

Collaborative design is a core element of engineering activity. While project-based work is often conducted by dividing tasks so that team members concentrate on their own part, collaborative design requires joint effort to work on the same drafts and ideas iteratively [8]. In engineering education, students' collaboration competencies could be promoted by especially emphasizing collaborative aspects in the team work. Collaborative learning is an educational method where students work in small teams to define, carry out and reflect upon a shared task, which often is a real-life problem. This approach emphasizes working on common tasks, sharing of responsibility and developing a working culture that promotes the usage of various perspectives and methods in developing outcomes together [9]. For instance, specific *New Product Development* (NPD) courses are an example of efforts to train engineering students together with the students from other technical areas and business training programs to manage challenges of multidisciplinary design [10–11].

From an educational point of view, the aim of collaborative design projects for students is not only to learn how to solve practical problems but also to understand and be able to articulate why a particular solution is supposed to work [12] . Typical challenges of collaborative design include an insufficient or inappropriate analysis of the design problem [13–14], design fixation and a preoccupation for a particular solution [15–16], as well as tendencies to reduce the complexity of the problem in order to arrive at a solution as soon as possible.

Present day engineering students will most likely work in partially distributed teams during their careers, as network mediated collaboration becomes more common. Successful implementation of collaborative practices requires software tools that support spatially, socially and temporally distributed participation in knowledge creation processes. Current information and communication technologies (ICT) are quite capable to support various kinds of knowledge-intensive work activities. However, the majority of learning technologies, such as virtual learning environments or groupware platforms (e.g., Moodle, BSCW or Blackboard) that are commonly used in education, are focused on the management and delivery of learning resources. These kinds of environments, which basically reify the needs and structures of higher education institution [17], focus on teacher-driven activities, limiting students' options for creative and self-organised participation. Current learning environments provide only limited support for knowledge building activities such as project management, sustained discourse and joint elaboration of resources, or boundary crossing collaboration [18–19]. For collaborative work, users are typically provided only with a storage space where users can upload and save documents. The documents remain isolated in the system, without possibilities for users to explicate and describe relations between them or tools for their joint development. Furthermore, these systems are usually closed entities and access rights are often determined by organizational constrains rather than by the needs of user groups.

A few existing educational applications have specifically been designed for collaborative knowledge creation, like Knowledge Forum (KF) for knowledge building practices, or FLE3 for progressive inquiry practices, but they start to be technically somewhat outdated and do not provide support for process coordination, social communication or networking integrated with the tools for managing knowledge objects [20]. In recent years, various new tools have been developed to support collaborative work and learning in general, as well as collaborative design in particular, but these tools either make little or no use of recent advances in semantic web technology or they are overly complex to use and hardly in line with the pragmatic requirements of students and professionals [21]. For instance, each of the various social software applications, nowadays freely available in the Web (wikis, weblogs, social bookmarking, social communities like Facebook, microblogs like Twitter, sharing applications like YouTube or Flickr etc.) provide affordances for some aspects of collaborative knowledge creation, but they are not easy to use in an integrated manner due to poor interoperability.

Real knowledge advancement and knowledge creation is a long-term process, in which progress takes place through series of inquiry and elaboration cycles. The challenge is to design an environment that supports such long-term and complex work also across individual courses or projects. To answer this challenge we present a web-based collaborative working and learning system, Knowledge Practices Environment (KPE), which is produced in the Knowledge-Practices Laboratory (KP-Lab) project [22] funded by the European Commission. The general objective of the technological research and development project was to design and implement a modular, flexible, and extensible ICT system that enables open and flexible linking and management of various elements in a complex knowledge creation process and supports collaborative project based methods in educational and workplace settings. The system is designed to provide tools for collaborative work around shared objects of activity and draw on the potential of emerging semantic web technologies.

2. The knowledge practices environment

The Knowledge Practices Environment (KPE) is a web-based virtual environment that provides a set of integrated tools and functionalities for sustained collaborative working with shared epistemic objects (artefacts, processes, practices).

The user environment is a virtual *shared space* (Fig. 1)—a graph-based view on collections of knowledge objects, which provide real-time and history based awareness to facilitate multi-user collaboration. Shared spaces can be used either individually or in cooperation with others. Collective spaces can be formed around a group of people belonging, e.g., to a project team, students attending a class, students of a university department, or any other type of collective.

The environment provides users with flexible means to import, create and modify shared objects (such as documents, task descriptions and other knowledge artefacts) as well as to organize them spatially according to different perspectives. Groups can assign resources and tasks to specific members and are provided with real-time information about activities of other users that are online at the same time. The technology builds on emerging technologies, such as semantic web, real-time multimedia communication, ubiquitous access using wireless devices, and inter-organisational computing.

Table 1 depicts the relations between KPE *views* and *tools* and the four types of user activities [20] as follows:

- Planning, organizing and coordinating processes;
- Creating, transforming and elaborating knowledge artefacts;
- Managing social relations around shared objects;
- Making visible and reflecting the work.

KPE *views* enable viewing the knowledge objects and their relations from different perspectives in a shared space. There are four main views to support the first three types of user tasks, namely the content view, the tailored view, the process view and the community view.

Various tools are integrated in these main views to enable object-bound development as well as flexible connection and organization of all information related to the knowledge objects, processes and

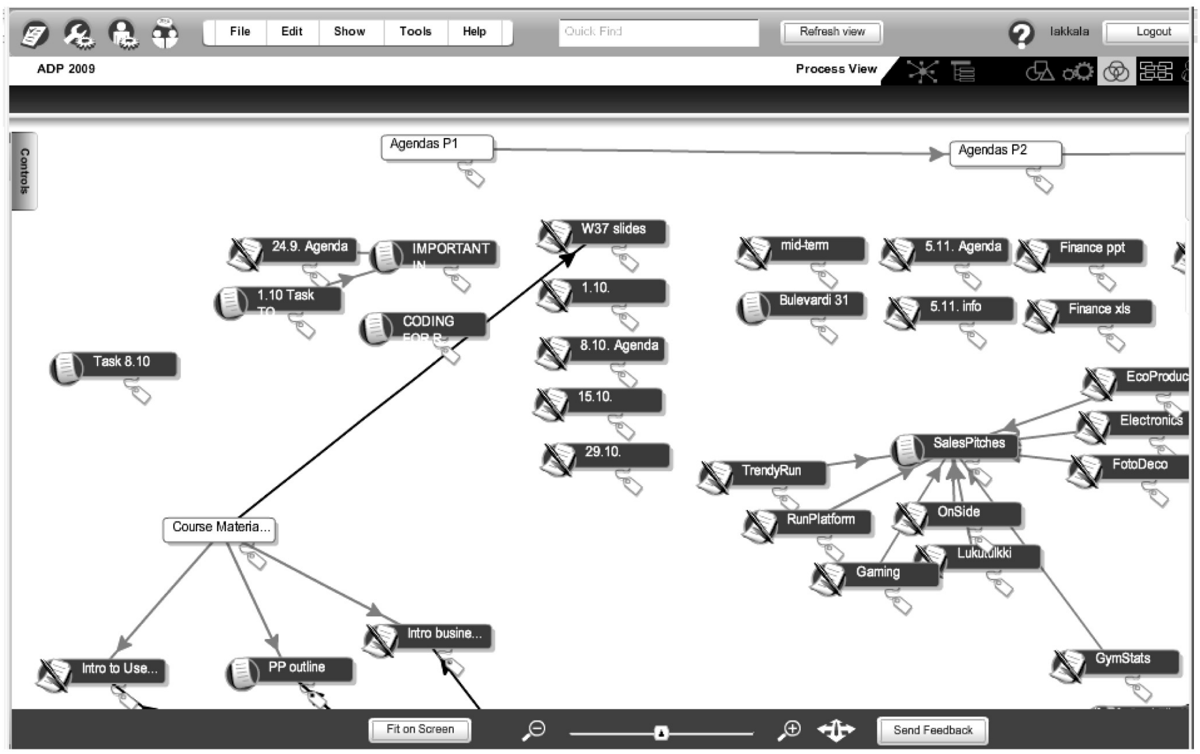

Fig. 1. KPE Shared Space (Content view).

Table 1. Main tools and features of the KPE and their relation to the user activities

Types of user activities	Planning, organizing and coordinating working processes	Creating, transforming, organizing and linking knowledge artefacts	Managing social relations around shared objects and linking people	Making visible and reflecting on the work processes
Shared space views and tools	Process view, Content view ToDo list, Gantt chart, Calendar, Meeting support, Chat	Content view, Tailored view Knowledge artefact management and linking, Tagging, Notes editor, Sketch pad, Wiki, Google docs, Visual models and languages editing	Community view Groups and roles management, mailing lists	Specialized views Timeline-based analyzer, Data export, Visual analyzer
Supporting features	Real-time synchronization (shared space content and user's presence), user's preferences, help, search, clipboard, authentication and authorization, activities logging.			

people concerned. In addition, KPE provides a supporting infrastructure, with features common to all views and tools—real-time synchronization and locking (essential for collaborative work on shared objects), awareness, logging, search and help.

2.1 Overview of KPE tools and views

In this section we will describe some of the more important KPE instruments, in the context of the user activities they support. The KPE provides both generic tools, available in multiple views (e.g. the Chat tool) as well as specialized tools (like the Gantt diagram editor) that belong to a single view.

2.1.1 Creating, transforming and elaborating knowledge artefacts

The primary 'environment' for working with knowledge artefacts is the shared space content view (Fig. 1). KPE content view supports the following artefacts types: content item, vocabulary, visual language, and visual model. The content item can be a file (uploaded from the user's workstation), a text document (note), a wiki page, a simple drawing or a hyperlink to an external web resource. *Vocabularies* contain lists of terms, used for tagging the knowledge artefacts. KPE supports both fixed and user-defined tag vocabularies. *Visual models* are graphical representations of objects and systems in terms of domain-specific *visual languages*.

Important functionality of the content view is the ability to spatially organize the knowledge artefacts on the screen and to define relationships (*links*) between them. When creating a link, a user specifies the relationship type by selecting it from a predefined taxonomy or by defining a new type.

Examples of tools facilitating document centred collaboration are the *Note Editor* (Fig. 2) [23] and the *Sketch pad.* The *Note editor* supports quick brainstorming and production of ideas and allows viewing, editing versioning, comparing and printing of shared documents in a flexible manner. The *Sketch pad* enables easy in-context drawing to support brainstorming and externalizing of ideas that are sometimes hard to explicate by only using verbal means.

The *Visual Model Editor* (VME) tool [24] allows users to create and elaborate visual models in the form of two-dimensional graph diagrams, such as flow-charts, argument-graphs, organigrams, decision trees, program logic models or conceptual maps both individually and collaboratively. The *Visual Modelling Language Editor* (VMLE) allows users to create, share and edit the visual modelling languages as such, thereby providing users with the possibility to create their own domain specific ontologies. The direct integration of VME into the shared spaces of KPE allows for an easy transition between modelling and other collaborative activities.

The visual models are represented as directed multigraphs. Each model is based on a particular modeling language and is constructed from the concepts and relationships, defined in this language. Fig. 3 depicts the VME user interface. On the figure, the visual model elements can be identified as white rectangles with arrows between them. With the VME, users can create references from the model elements to the knowledge artifacts in the related shared space (shown as darker rectangles in the background in Fig. 3). Fig. 4 shows an example of modeling language developed using Visual Language Editor, with the concept and relationship hierarchies clearly identifiable [25].

One digital tool for visual modelling and conceptual mapping, widely used in educational contexts, is the IHMC CmapTools software [26–27], which has many similarities with KPE and VME integrated with it. However, CmapTools have some clear restrictions and shortcomings compared to KPE and VME. For instance, unlike VME, it constrains the direction of modelling process, which can be seen to favour hierarchically structured concept maps. Hierarchical structuring is widely used in representing static relationships between concepts,

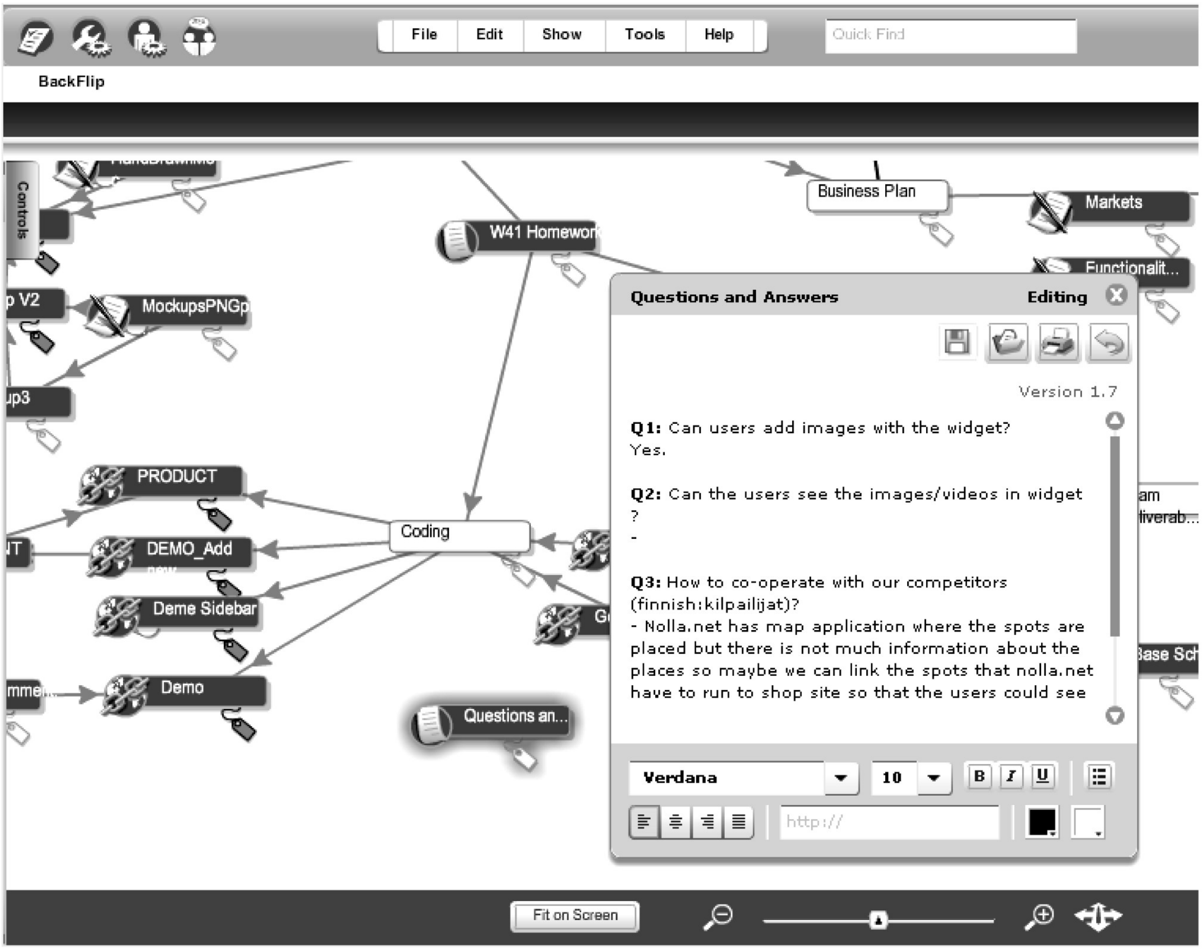

Fig. 2. Note written with the Note Editor by a student team to collect questions concerning their design task.

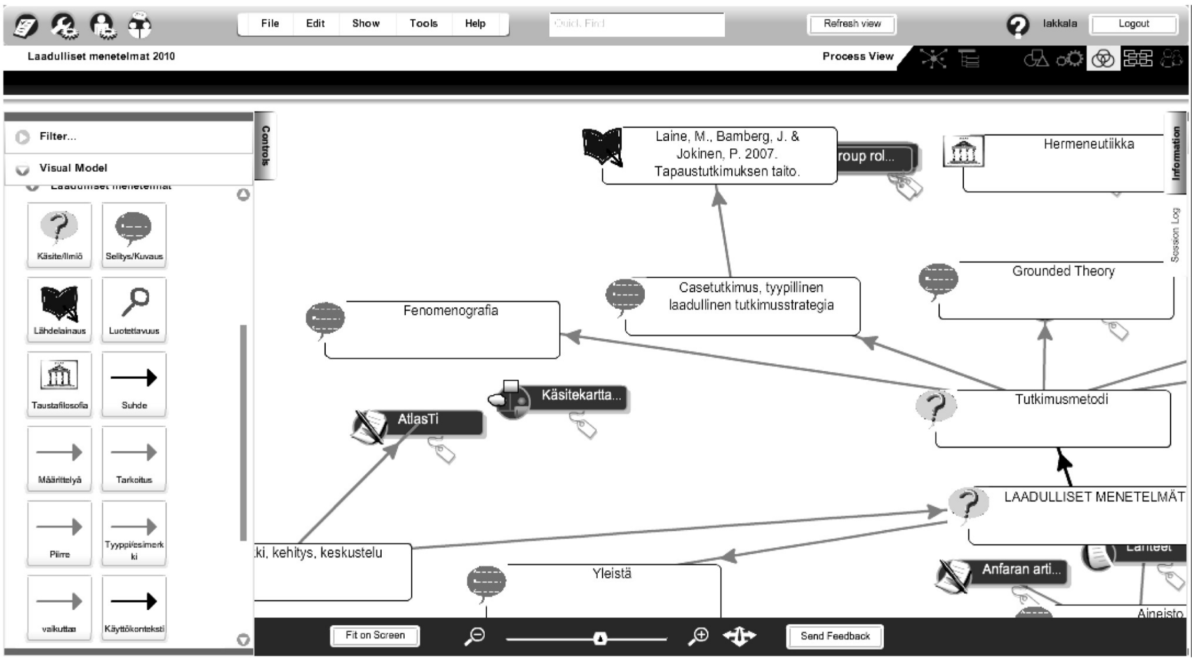

Fig. 3. A visual model created by a group of university students with VME, using a visual modelling language defined by the teacher.

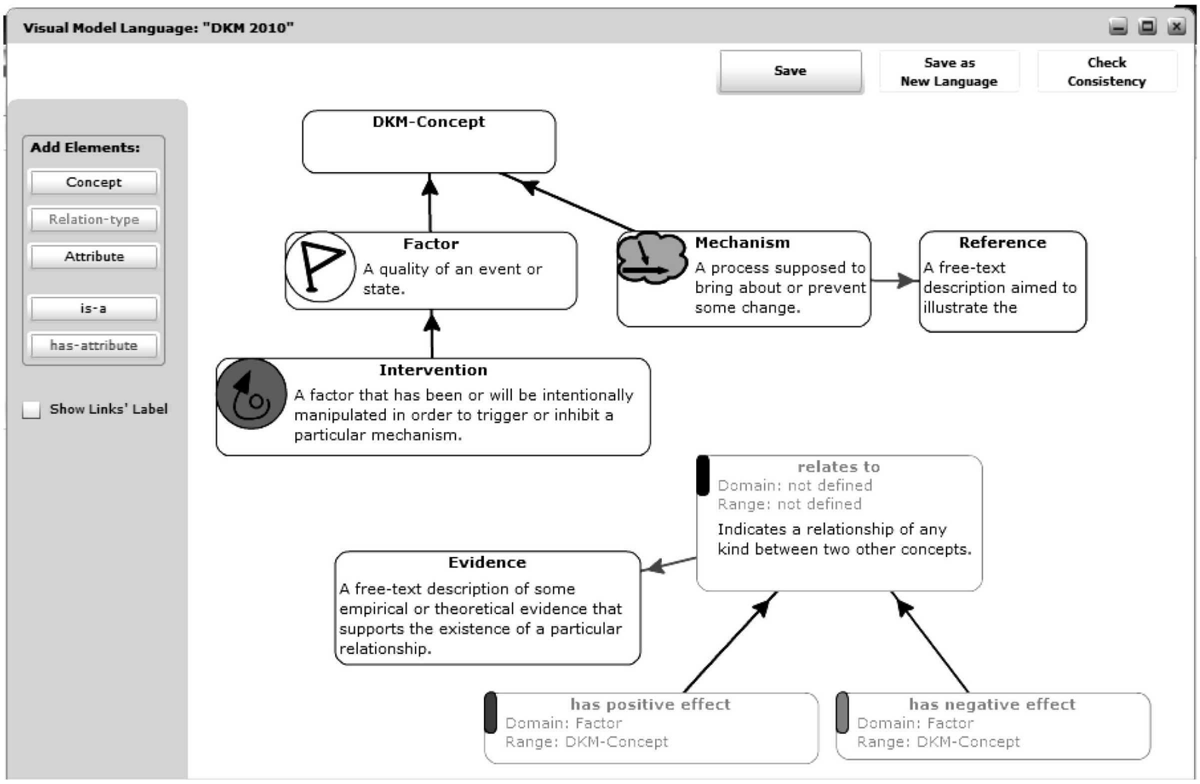

Fig. 4. Visual Modelling Language Editor.

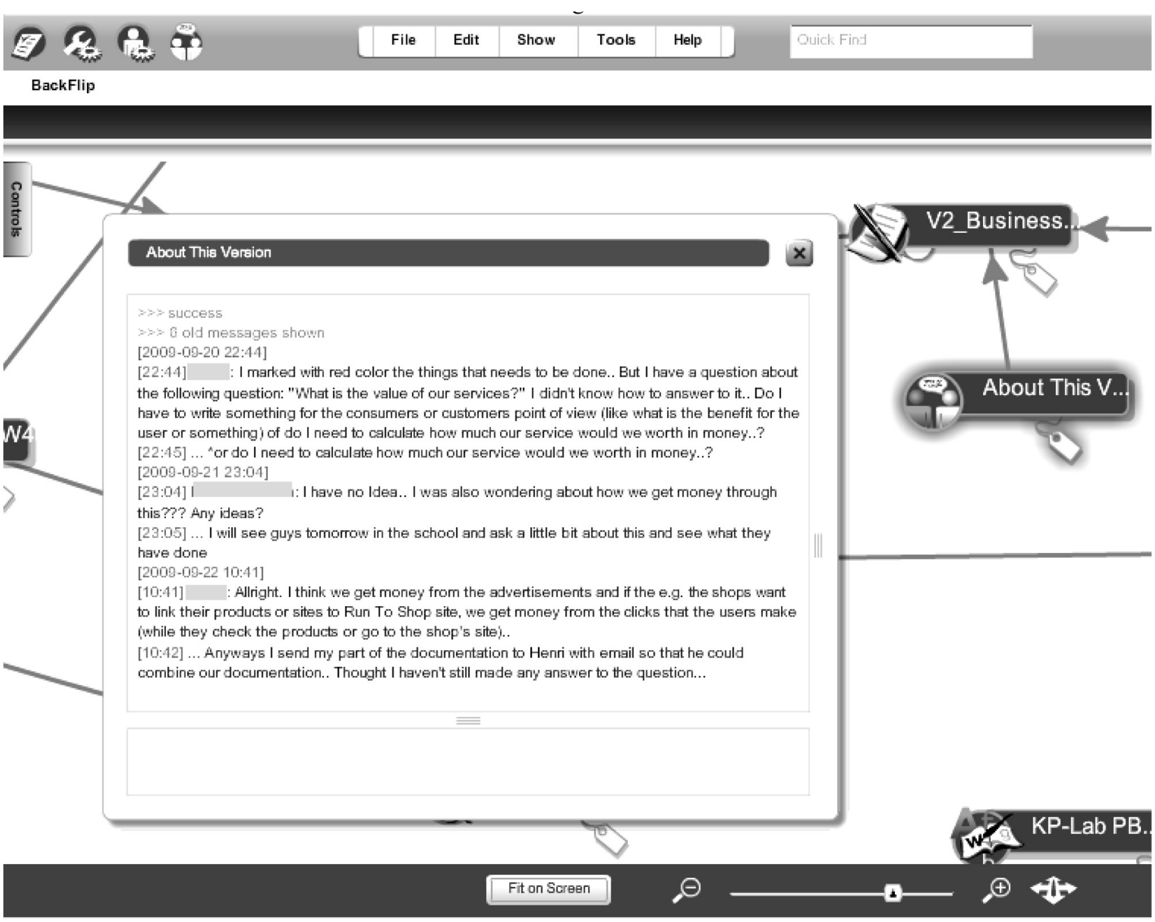

Fig. 5. An example of a saved object-bound Chat session by a student team discussing their business plan in an engineering course.

but may not optimally support the modelling of functional relations, dependencies and sequential content, such as processes, time-lines or developments [28–29]. CmapTools affords linking of external resources into the conceptual maps by means of exporting functionality (a user can copy external files or web-links to be presented as hyper-links within separate nodes of a map), but it does not support the creation of parallel links between separate resources and multiple concept maps, since a user has to export resources separately into each created map. KPE and VME, instead, allow users to create their maps in an integrated fashion on a separate layer while the used shared space beneath remains visible. The separate elements of conceptual maps can be linked to any digital items uploaded into the used shared space, and a separate digital resource can be linked to many nodes and models. The Visual Modelling Language Editor (VMLE), integrated with KPE, corresponds to the Concept-map Ontology Environment, built on top of the IHMC Concept Map Tools tool suite. However, unlike the former tool, VMLE allows a user to endow modelling languages with default link colours and exported pictures included into the node elements.

2.1.2 *Planning, organizing and coordinating working processes*

Tasks and *process entries* are used to describe workflows, assign responsibilities and define deadlines in a shared space of KPE. The project teams are encouraged to use them for planning and organizing their work. In the *process view* of a shared space, tasks and process entries can be viewed and organized using a traditional Gantt diagramming tool. A special feature of KPE is that the same task items are visible and modifiable also in the *content view*, where they can be presented, linked and arranged with the content items of the same space.

The *Chat tool* [30] is often used by the project team members to conduct focused *object-bound* discussions (Fig. 5). The capability to link the chat to a particular knowledge artefact facilitates the contextualized synchronous user communications when working on shared objects. Chat histories are considered to be important project artefacts and are stored in a database.

2.1.3 *Managing social relations around shared objects*

Community view shows the team members and allows defining and arranging *groups* (Fig. 6). The links between persons and groups represent either an affiliation or role (e.g. 'is mentor of', 'is test user of'). The numbers of groups and their types as well as the members' roles in each group can be defined by the shared space members themselves. Also the responsibilities and contributions of each member of a shared space in creating and modifying shared items throughout the space lifetime can be examined in the community view.

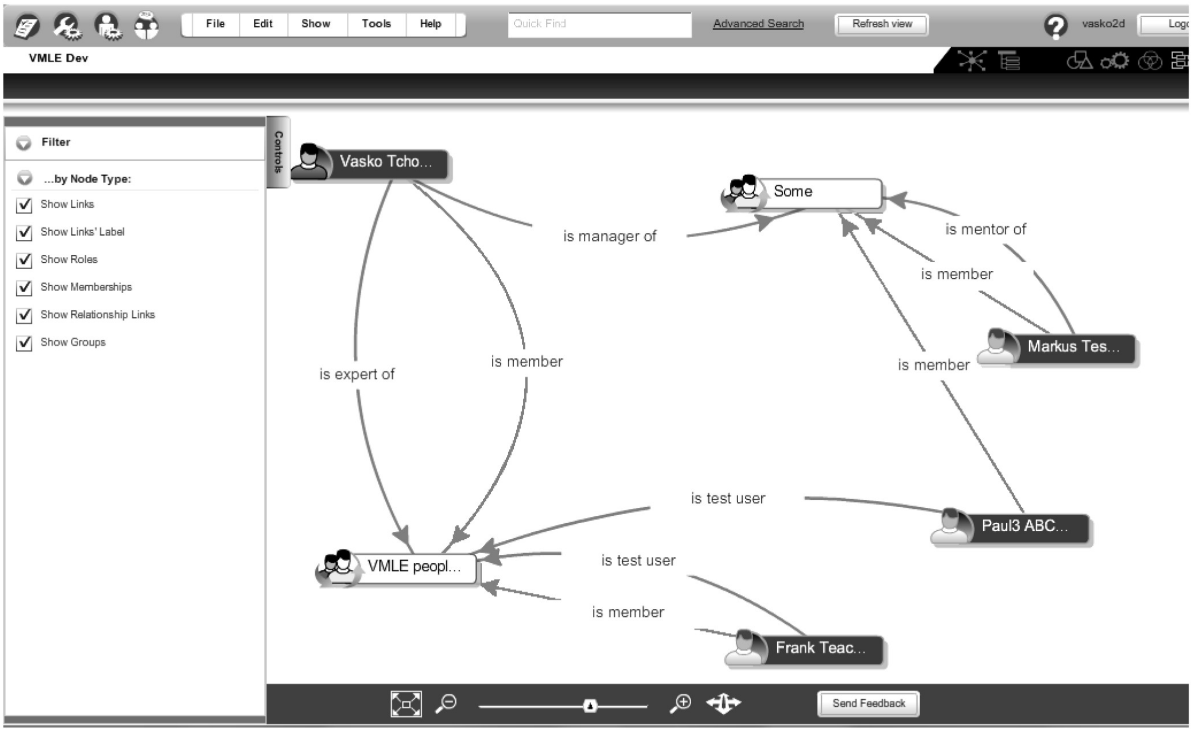

Fig. 6. Community view.

2.1.4 Making visible and reflecting the work

During a project development process, all user actions that modify the state of the knowledge artefacts are recorded by the KPE system. A specific cluster of functionalities, so called *analytic tools* are available in KPE to facilitate teachers, students and researchers in analyzing this information and identifying patterns in conducted collaborative activities around shared objects. Analytic tools include such applications as Data Export Tool, Visual Analyzer and Timeline Based Analyzer, which process data from user action logs according to the query parameters selected by a user and convert processed data into concise texts, tables and visualizations. These representations allow users to monitor and reflect on their collaborative work, including contributions of individual members on separate content items and other forms of participation, and the intensiveness of the work on various content items during the selected time period.

2.2 KPE Architecture and technical implementation

The Knowledge Practices Environment is a web application (Fig. 7), which consists of user client, distributed services and data repositories.

The client is implemented in Adobe Flash. When the user initiates a KPE session, the client is loaded in the web browser, and the system provides a graphical or list view of all shared spaces in the server. By opening a shares space, a user sees a visual representation of the shared space content (Fig. 1).

The client collaborates with various KPE services:

- Knowledge services (Knowledge Browser, Knowledge Annotator)—manipulation, retrieval and persistence of shared objects and links between them;
- Content management service (aka Content Item service)—management of digital artefacts associated with the shared objects;
- Tool specific services (visual modelling languages, data import/export, time-based analyzer)
- Search service—shared object search engine, based on Apache Solr/Lucene [31];
- Logging service—via this service, KPE tools can store records about user activities for later introspection and analysis;
- User authentication—part of the KPE *single sign-on* infrastructure [32];
- Synchronization service—keeps the client UIs in sync. All users of a particular shared space can see the changes in the shared space content and layout, inflicted by other users, almost simultaneously.

KPE data storage infrastructure consisting of:

- Content repository—based on Apache Jackrabbit [33]—for storing the digital artefacts associated with the shared objects (e.g. uploaded files, notes, sketches, object-bound chat logs);
- RDF repository [34]—for keeping meta-data (roughly based on Dublin Core [35] about the shared objects and the links between them;
- Tool-specific repositories—wiki, visual modelling languages, search.
- Service databases—logging, user authentication.

Most KPE services are implemented as JAX-WS [36] based web services, deployed on Oracle GlassFish [37] and Apache Tomcat [38] application servers. This architectural decision allows for a truly

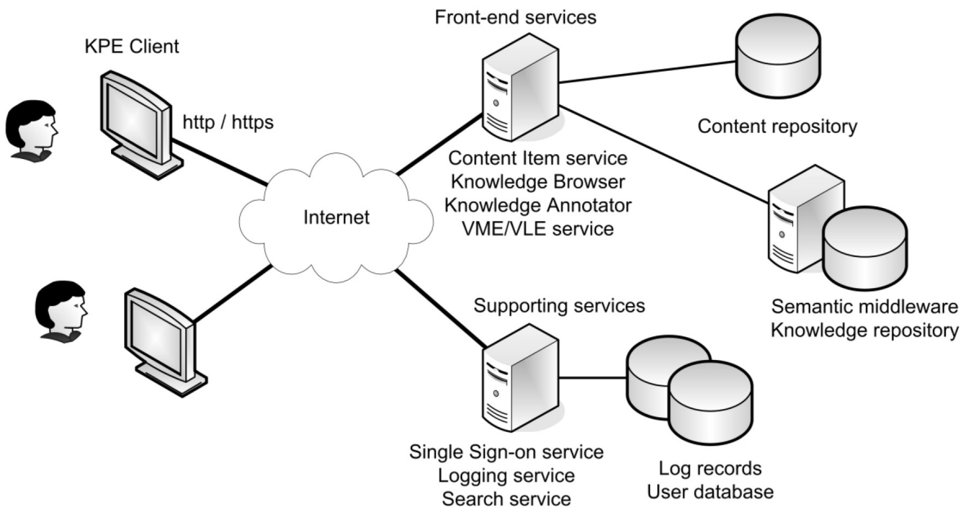

Fig. 7. KPE Architecture.

distributed KPE deployment with different services residing on different hosts.

3. User experiences from pilot studies

As part of the research in the KP-Lab project, successive releases of KPE was used and investigated in several higher education courses related to project work and knowledge creation practices. Below the main results of some of these design-based research endeavours concerning the technological affordances are described.

In Helsinki Metropolia University of Applied Sciences in Finland, KPE was used in application design courses for engineering students. During the spring 2008, the usage of an early version of KPE was investigated in a Multimedia Product course for second-year bachelor level students [38]. During the course, students designed multimedia products in teams for real customers, and shared their design documents and tasks through KPE. The main benefits of KPE, reported by the students, was that the user interface with space-like views afforded flexible management of knowledge resources better than typical folder-based environments. However, visual complexity was also reported as a problem if the number of items in a space increases. From the pedagogical point of view, the analysis of student teams' collaboration practices revealed that the engineering students easily adopted a practice were tasks are divided between team members and the affordances of the tools for collaborative coordination, mutual commenting or co-editing of shared objects were not exploited as much as was expected.

In another study [39], a more mature release of KPE was used in an 'Application Development Project', in which students from engineering and business training programs developed business ideas and related multimedia applications for real customers. The students worked in teams and they were supposed to iteratively advance their business ideas, and create concrete business plans and technical solutions on the basis of these ideas. The students were also required to take care of the workflow in the teams by planning and coordinating their activities and defining the amount of time allocated for separate tasks. The development of ideas, initial solutions and mock-ups required the students to manage numerous memos, drafts, documents, URL-addresses and external knowledge resources through KPE. According to the results, KPE was found to afford a) the organisation of various documents and other items into functional clusters, b) the commenting of separate documents and tasks, and c) the easy creation and flexible modification of separate coordination artefacts, e.g., by the Note editor, allowing the students to operatively divide the tasks and responsibilities in the teams during the separate phases of their joint work.

Such functionalities of KPE as uploading files, creating notes and web-links as well as connecting the created content items with graphical links were mostly used to organize the work on various versions of documents and mock-ups. The use of KPE allowed the student teams to organize their subtasks visually as well as explicate the sequential order of versions and interdependencies between diverse intermediate documents. One of the student teams in the investigated course used link items in explicating multiple connections between the coordination artifacts and various tasks (Fig. 8) and documents. When the team was asked about their way to implement the linking functionality, one of the team members said:

> It's better than in folder sharing in Google Docs or Drop-Box. Here you can see what relates to what and you can really manage those things and it's quite logical thing. It helps when you are searching for certain file which is made on same day or which is related to some other document. You can find it faster.

The same team used also the object-bound Chat in discussing and commenting on separate documents and tasks. According to the team members, one of the advantages of KPE is that it supports commenting and discussing focused on particular files, thus allowing users to discuss in more detailed way, while in the other systems, as a rule, only one, isolated discussion board is available. Chat editor was considered helpful because it enabled also discussions taking place in a synchronous fashion.

The first release of the Visual Model Editor was tested in a seminar about qualitative research methods in the University of Helsinki. According to the experiences [40], the students actively explored with the functionalities of VME and suggested ideas about improving them. One central wish was to have links from the concept maps to other resources in their shared space related to the concepts. In the subsequent year, the same seminar was conducted again with more mature versions of KPE and VME, including also the possibility to link the elements of the model to other resources in the shared space.

In a case study [25], VME was tested in undergraduate design courses in the University of Applied Sciences Upper Austria. Throughout a period of five months, 35 students in 10 project teams were assigned to envision, develop, and assess a solution for complex design problem in the fields of eCommunication and eModeration. KPE and VME were introduced to the students for explicating their evolving conceptions of their team's design problem

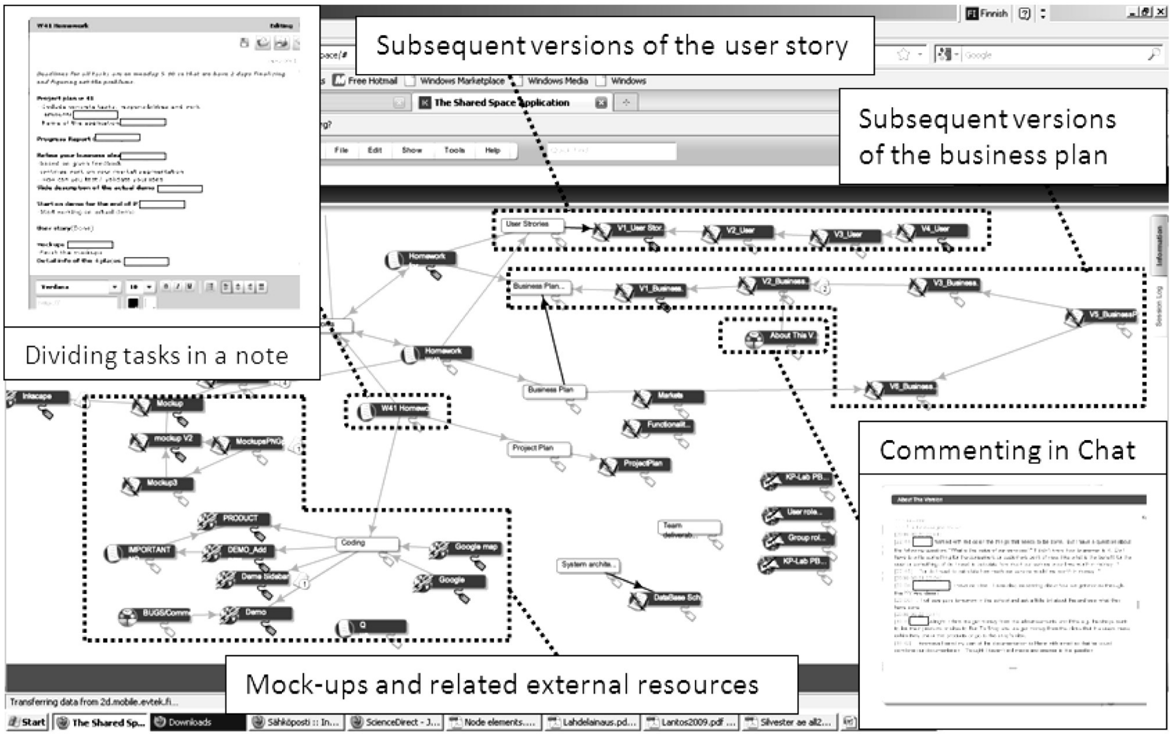

Fig. 8. An overview of the shared space and it items of one student team in the Application Development Project.

and solutions. A simple modelling language was introduced to scaffold the work. The main results reported by the authors were that student teams adopted the new tools in a varying degree and the utilization of semantically rich conceptual models seemed to depend on the direct added value experienced by the users. For instance, some student groups hardly used VME, when others revised their models frequently and created several different models. One result concerning the pre-defined visual modelling language was that it was too restrictive and students had difficulties in mapping their ideas to the provided concepts. This result indicated that users need a possibility not only to edit the models but also the underlying modelling languages. Later that possibility has been integrated in KPE and VME through the Visual Model Language Editor, which was not yet available at the time of the study.

In a Finnish higher-education undergraduate course, organized as a cross-curricular setup between three universities, theories, methods and practices of multidisciplinary, virtual project work have been taught for technical, business, and psychology students. During the course, student teams construct theory-informed service solutions for real customer organizations. In the previous iteration of the course, a conventional virtual learning environment (Optima [41]), offering a virtual workspace with hierarchical folder structures, file sharing and discussion forums, was used for managing collaborative activities and materials. According to the research results from the course [42], the Web-based environment was experienced improper because of difficulties in keeping track on each team's material and finding right areas; it was mainly used for storing documents and coordinating team work in discussion forums. The authors concluded that more flexible Web-based technology could provide more versatile support for a team's collaboration, including pragmatic, social, epistemic, and reflective types of activities.

The most recent iteration of the same course ('Advanced Themes on Project Management') was conducted in the spring 2010 with the same pedagogical design but using KPE as a virtual project management tool. About 30 students took part in the course. During the course, students were asked to analyze the peculiarities of different types of taxpayers and to create suggestions for the Finnish Tax Administration based on the analysis. The course was investigated as one central research case of the KP-Lab project. The rich and multi-faceted data (lesson observations, interviews, database content etc.) was investigated using a mix method approach with various quantitative and qualitative methods. Below the main results from the point of view of technology usage are summarized.

According to the results, the investigated six student teams were found to have used such KPE functionalities as Note editor, uploading files, creation of task items, linking between items, commenting and chatting. The Note editor was widely implemented in various epistemic activities of the student teams. Thus, four out of six teams used the Note editor in brainstorming about the topics of their projects and the content of project plans (see Table 2). The team that had most actively used KPE during the course, explained in their interview that the joint drafting of notes in preparing the final presentation helped them to integrate all ideas together and then split the whole task into subtasks for each member to work on. The majority of the teams also shared knowledge resources in their working spaces. In two team spaces, Note editor was implemented in creating literature reviews while three of the teams only uploaded research papers and other resources as files into their working spaces. All the teams uploaded various versions of the working documents that they created during the course into their shared space.

The student teams of the course came up with similar ways of using the Note editor and chatting for planning, regulating and coordinating their joint activities in the teams, as were found in the Application Development Project [39]. The majority of the teams used task or link items in structuring their working spaces and assignments being worked on (see Table 2). One team also frequently used the object-bound chatting functionality. One member from the team that most actively used KPE during the course stated in his interview that KPE appears to support open-ended working process, allowing users to initiate new unforeseen branches to work.

Analytic tools were also somewhat tested for the first time in the Advanced Themes on Project Management course. A member of the student team, responsible for analyzing the course activities, used the Data Export Tool to analyze individual members' actions on their team's content items. In addition, one of the instructors made use of both the Visual Analyzer and the Timeline Based Analyzer to compare the teams' activities as well as separate team members' engagement in the activities mediated by KPE. In the post-interview, the instructors emphasized the potential of analytic tools to follow activities related to specific documents, a feature not supported by other web-based collaboration tools.

4. Discussion

In the previous section, we described the main results of case studies, conducted during the KP-Lab project, in which KPE was implemented in some pilot courses where students were taught project work and collaborative design practices. According to the experiences from the courses, KPE implemented in project based learning activities may provide present-day students with new possibilities to experiment with object-oriented, virtual collaboration practices that will be common in their future working life, such as the following:

Self-directed knowledge advancement—as a project involves multidisciplinary knowledge, which is not covered by standard lecture material, students are driven to study and seek multiple solutions and combine various viewpoints, conceptualizations and pieces of information, which serve to enhance their understanding of theoretical and professional knowledge. One powerful feature of KPE, emphasized by many pilot users, appears to be the combination of tools and functionalities in the Content view that allows flexible co-construction, linking and arrangement of complex knowledge spaces;

Table 2. Summary on the usage of KPE functionalities and various uploaded digital artefacts during the 'Advanced Themes on Project Management' course 2010

	Team 1	*Team 2*	*Team 3*	*Team 4*	*Team 5*	*Team 6*
Usage in epistemic activities						
Brainstorming about ideas and content	8 notes	1 note	2 notes	2 notes		
Sharing and reviewing knowledge resources	3 files 1 comment	1 file	1 note	2 notes 7 files		1 file
Working on documents	5 files	5 files	5 files	19 files	4 files	2 files
Usage in regulation and coordination						
Defining rules, tasks and responsibilities	1 task item 1 file	7 task item 3 files	2 task item 2 files	7 task items 4 notes 2 files	1 task item	
Groups of items united with links	2	6	1	4	1	
Negotiating informing	1 file 4 comments			1 chat item		
Reflecting on activities	1 note	1 file	1 note	4 notes	3 files	

Project management—students have to plan, organize and monitor the design task by themselves, dividing responsibilities and coordinating their collaborative activities through KPE against process progression and a specified timeline. However, different student teams in the investigated courses adopted the new tool in varying degree, and only some teams found the unique possibilities and added value of the tool functionalities for effective project work.

Communication and collaboration competences—KPE facilitate user interaction in a contextualized manner, supports quick brainstorming and production of ideas together, networking and community building through synchronous and asynchronous user communications when working on shared objects. Social mediation provided by the tools allows users to lean on each others' competencies, expertise and experience and help them align their actions with those of others. Awareness features include clues and notifications of participants' status or past and present activities. KPE makes explicit and visualizes participants' activities in the virtual spaces, which seem to help students become more conscious with the challenges and systematic with the strategies of collaborative knowledge work.

A functionality that has not yet been available for end-users, but is planned to be implemented, is the integration of Google Docs access from shared spaces. The integrated access allows end-users to work on Google Docs documents with the KPE password and enter to Google Docs environment directly from the KPE Content view, enabling easy linking of the documents made by Google Docs with other shared knowledge resources. The use of Google Docs in an integrated fashion will expand the pedagogical potential of KPE since the tools of Google-docs support sustained, collaborative co-construction of large texts, whereas the KPE Note editor is more suitable for quick brainstorming and co-editing of short notes and drafts.

Also the analytic tools have not yet been released for a wide range of end-users. In the pilot courses investigated in the KP-Lab project, the analytic tools have mainly been used by researchers and teachers to monitor and analyze the progress of the process, but the ultimate pedagogical goal is to introduce them to all students and teach them to self-reflect their collaboration practices. However, more experiments of using the analytic tools by end users are needed in order to evaluate their usefulness in educational practices using PBL.

5. Conclusions

The paper presents the Knowledge Practices Environment, a web-based flexible and extensible environment with a cluster of interoperable applications to foster collaborative knowledge creation and continuous work around shared objects. The environment makes use of recent Web 2.0 and Semantic Web applications and provides a coherent set of tools in order to initiate and organize collaborative processes, to support asynchronous and synchronous communication among participants, and to create, work on, share, and organize documents and artefacts collectively.

KPE is aimed to support students, teachers and professional practitioners in their activities, in which the aim is to engage in practices of collaborative knowledge work and improve related competencies of the participants. In the KP-Lab project, the prototype of KPE was tested in several higher education courses, in order to evaluate whether the original design objectives of the tool could be met in educational practice. KPE was designed to presents a novel way to support collaborative knowledge practices through providing flexible integration of various aspects of a knowledge creation process as well as multiple views to organize shared activities and knowledge artefacts.

According to the results of the pilot courses, users value the new possibilities provided by KPE to manage the entire collaborative process and shared knowledge spaces through modifiable and expandable spatial views. Also the possibilities to examine the same objects through multiple views, create links between various aspects of the collaborative endeavour and follow the process progression in various ways were experienced as important affordances of the new tool. However, it is also important to realize that a digital tool alone cannot bring about qualitative improvements towards more advanced and effective knowledge work practices. Spontaneously, individual students and student groups seem to appropriate the new affordances in varying degree and quality. Educational practitioners willing to take KPE in use in their teaching need to acknowledge that its usage has to be embedded in pedagogical aims, structures and guidance that direct students to use the affordances of the tool functionalities in a way that is beneficial for learning collaborative knowledge creation practices and related competencies. In future studies, it is important to get experiences of the usage of fully functional and complete KPE system complemented with appropriate pedagogical design and guidance in various educational settings.

Acknowledgements—The reported work is carried out within the KP-Lab project funded by the EU 6th R&D Framework program. We thank all our colleagues and partners involved in the design, implementation, and evaluation of the Knowledge Practices Environment and integrated tools.

References

1. N. Stankovic, Single development project. *The Journal of Systems and Software*, **82**, 2009, pp. 576–582.
2. H. Muukkonen, M. Lakkala, J. Kaistinen and G. Nyman, Knowledge creating inquiry in a distributed project management course, *Research and Practice in Technology-Enhanced Learning*, in press.
3. W. D. Nance, Improving information systems students' teamwork and project management capabilities: Experiences from an innovative classroom, *Information Technology and Management*, **1**, 2000, pp. 293–306.
4. Bologna Process, http://www.ond.vlaanderen.be/hogeronderwijs/bologna/ Accessed 30 November 2010.
5. Leuven/Louvain-la-Neuve Communiqué, The Bologna process 2020—The European Higher Education Area in the new decade, http://www.ond.vlaanderen.be/hogeronder wijs/bologna/conference/documents/Leuven_LouvainlaNeuve_Communiqué_April_2009.pdf, Accessed 30 November 2010.
6. C. L. Dym, A. M. Agogino, O. Eris, D. D. Frey and L. J. Leifer, Engineering Design Thinking, Teaching, and Learning, *Journal of Engineering Education*, **34**, 2005, pp. 103–120.
7. J. E. Millis and D. F. Treagus, Engineering education—is problem-based or project-based learning the answer? *Australasian Journal of Engineering Education*, 2003, pp. 18–30, on-line publication, http://www.aaee.com.au/journal/2003/mills_treagust03.pdf, Accessed 30 November 2010
8. E. Lehtinen, Computer supported collaborative learning: An approach to powerful learning environments, In E. De Corte, L. Verschaffel, N. Entwistle and J. Van Merriëboer (Eds.), *Unraveling basic components and dimensions of powerful learning environments*. Elsevier, Amsterdam. 2003, pp. 35–53.
9. P. Dillenbourg, *Collaborative Learning: Cognitive and Computational Approaches*, Advances in Learning and Instruction Series, Elsevier, New York, 1999.
10. G. P. Lantos, D. L. Brady and P. H. McCaskey, New product development. an overlooked but critical course. *Journal of Product Brand Management,* **18**, 2009, pp. 25–436.
11. K. J. Silvester, J. F. Durgee, C. M. McDermott and R. C. Veryzer, Perspective: integrated market-immersion approach to teaching new product development in technologically-oriented teams, *Journal of Product Innovation Management,* **19**, 2002, pp. 18–31.
12. A. H. Dutoit, T. Wolf, B. Paech, L. Borner and J. Ruckert, Using rationale for software engineering education,CSEET, *18th Conference on Software Engineering Education & Training*, April 18–20, 2005 Washington, DC, pp. 129–136.
13. J. R. Mathias, A study of the problem solving strategies used by expert and novice designers, PhD Thesis, University of Aston, Birmingham, UK, 1993.
14. C. Ho, Some phenomena of problem decomposition strategy for design thinking: differences between novices and experts, *Design Studies*, **22**, 2000, pp. 27–45.
15. A. T. Purcell and J. S. Gero, Design and other types of fixation, *Design Studies*, **17**, 1996, pp. 363–383.
16. M. Mähring and M. Keil, Information technology project escalation: a process model, *Decision Sciences,* **39**(2), 2008, pp. 239–272.
17. S. Wilson, O. Liber, M. Johnson, P. Beauvoir, P. Sharples and C. Milligan, Personal Learning Environments: Challenging the Dominant Design of Educational Systems. *Proceedings of the Joint International Workshop on Professional Learning, Competence Development and Knowledge Management*, LOKMOL and L3NCD, Crete, Greece, 2006, pp. 67–76.
18. M. Scardamalia and C. Bereiter, Computer support for knowledge-building communities, *Journal of the Learning Sciences*, **3**(3), 2003, pp. 265–283.
19. P. Sharma and S. Fielder, Introducing technologies and practices for supporting self-organized learning in a hybrid environment. *Proceedings of I-KNOW '04*, Graz, Austria, June 30–July 2, 2004.
20. M. Lakkala, S. Paavola, K. Kosonen, H. Muukkonen, M. Bauters and H. Markkanen, Main functionalities of the Knowledge Practices Environment (KPE) affording knowledge creation practices in education. In: C. O'Malley, D. Suthers, P. Reimann, & A. Dimitracopoulou (Eds.), *Computer supported collaborative learning practices: CSCL2009 conference proceedings,* Rhodes, Creek: International Society of the Learning Sciences (ISLS), 2009, pp. 297–306.
21. M. Van Kleek, M. Bernstein, P. André, M. Perttunen, D. Karger and M. Schraefel, Simplifying knowledge creation and access for end-users on the SW, Proceedings of *CHI 2008 Workshop on Semantic Web User Interfaces*, April 5, Florence, Italy, 2008.
22. ICT-27490 Knowledge Practices Laboratory (KP-Lab) Project, www.kp-lab.org, Accessed 30 November 2010.
23. I. Furnadziev, V. Thoumatchenko, T. Vasileva and L. Benmergui, Note Editor for Document Centred Collaboration, *Proceedings of World Conference on Educational Media, Hypermedia and Telecommunications*, *ED-Media 2008,* Vienna June 30–July 4, 2008, pp. 1995–2000.
24. C. Richter, H. Allert, V. Tchoumatchenko, T. Vasileva, L. Ilomäki and Kari Kosonen, Visual Model Editor for Supporting Collaborative Semantic Modeling, *V International Conference on Multimedia and Information & Communication Technologies in Education m-ICTE2009*, Lisbon, Portugal, 22–24 April, 2009, pp. 603–607
25. V. Tchoumatchenko, T. Vasileva, C. Richter and H. Allert, Tools for Collaborative Development of Visual Models and Languages, *The IEEE Engineering Education Conference EDUCON'2010* – The Future of Global Learning in Engineering Education, 14–16 April, Madrid, Spain, 2010, pp. 1597–1602.
26. C. M. Steiner, D. Albert and J. Heller, Concept mapping as a means to build e-learning. In: N. A. Buzzetto-More (Ed.), Advanced principles of effective e-learning, Santa Rosa, CA: Informing Science Press, 2007, pp. 59–11CmpTools, Available at http://cmap.ihmc.us, Accessed 30 November 2010.
27. M. J. Eppler, A Comparison between Concept Maps, Mind Maps, Conceptual Diagrams, and Visual Metaphors as Complementary Tools for Knowledge Construction and Sharing, In: *Information Visualization*, **5**(3), 2006, pp. 202–210.
28. F. Safayeni, N. Derbentseva and A. J. Cañas, A Theoretical Note on Concept Maps and the Need for Cyclic Concept Maps. *Journal of Research in Science Teaching*, **42**(7), 2005, pp. 741–766.
29. I. H. Furnadziev, V. P. Tchoumatchenko , T. K. Vasileva and M. H. Lakkala, M. Bauters, Tools For Synchronous Communications in Collaborative Knowledge Practices Environment (KPE), *V International Conference on Multimedia and Information & Communication Technologies in Education m-ICTE2009*, Lisbon, Portugal, 22–24 April, 2009, pp. 588–592.
30. Apache Solr, Available at http://lucene.apache.org/solr/, Accessed 30 November 2010.
31. JOSSO—Java Open Single Sign-On Project, http://www.josso.org, Accessed 30 November 2010.
32. Apache Jackrabbit, Available at http://jackrabbit.apache.org, Accessed 30 November 2010.
33. RDFSuite, Available at: http://www.kp-lab.org/tools/rdfsuit, Accessed 30 November 2010.
34. Dublin Core Metadata, Available at http://dublincore.org/documents/dcmi-terms/, Accessed 30 November 2010
35. JAX-WS Reference Implementation Project., Available at http://jax-ws.java.net/, Accessed 30 November 2010
36. GlassFish Server, Available at http://glassfish.java.net/, Accessed 30 November 2010.
37. Apache Tomcat Server, Available at http://tomcat.apache.org/, Accessed 30 November 2010.
38. S. Jalonen, M. Lakkala and S. Paavola, Investigating knowledge creation technology in an engineering course, 2010. A manuscript submitted for publication.
39. S. Paavola (Ed.), *DIV.6 Report on empirical research*. A report for the European Comission, KP-Lab Project, IST-2004-27490, 2010.
40. K. Kosonen, L. Ilomäki and M. Lakkala, Collaborative conceptual mapping in teaching qualitative methods. In: T. Joutsenvirta and L. Myyry (Eds.), *Blended Learning in Fin-*

land. Faculty of Social Sciences at the University of Helsinki, Helsinki, 2010, pp. 138–153.
41. Optima, Available at http://www.discendum.com/, Accessed 30 November 2010.
42. H. Muukkonen, M. Lakkala, J. Kaistinen and G. Nyman, Knowledge creating inquiry in a distributed project management course, *Research and Practice in Technology Enhanced Learning*, **5**(2), 2010, pp. 1–24.

Prof. Tania Vasileva received M.Sc. Degree in Electronics and Ph.D. in Microelectronics from the Technical University of Sofia, Bulgaria. She currently works as an Associate Professor in Department of Electronics at the same University, teaching Semiconductor Devices, ASIC and VLSI Design. Her research interests are in VLSI design, as well as computer aided design tools and analysis. She is also interested in introducing collaborative knowledge practices in electronics engineering education. She has participated in several national and international research and development projects concerning methods and practices of using information and communication technology in higher education, interactive tools for distributed and collaborative learning, virtual environments and remote labs in engineering education. She has published over 150 technical papers, 8 technical books, and 19 interactive multimedia training tools for e-learning in electronics.

Dr. Minna Lakkala, Ph.D. works currently as a project researcher at the Technology in Education Research Group (TEdu), Institute of Behavioural Sciences, University of Helsinki. Her background is in general psychology, computer science and sociology. She made her Ph.D. about teachers' practices in scaffolding and organizing students' technology-enhanced collaborative inquiry. She has participated in several national and international research and development projects concerning, e.g., school development through ICT, students' and teachers' ICT skills, the pedagogical design of technology-mediated knowledge practices, and digital learning materials. She has also worked several years as a trainer of in-service teachers in the areas of computer aided instruction and advances of educational practices with modern technology, and as a consultant of various work organizations concerning the changes of learning and working practices with technology.

Vassiliy Tchoumatchenko, Ph.D, received his M.Sc. Degree in Radio Engineering from the Technical University of Sofia, Bulgaria, and Ph.D. degree in Microelectronics from INPG, Grenoble, France. He works currently as a senior assistant professor at Department of Electronics, Technical University of Sofia with teaching activities in the field of ASIC and VLSI Design, and Computer networking intensively using Project based learning and practices for collaborative knowledge creation. His research interests are in software engineering and CAD of integrated circuits. He has participated in many national and international research and development projects dealing with development of virtual learning environments, software tools for e-learning, collaborative learning, and virtual electronics laboratories. He has published over 90 technical papers, 4 technical books and interactive multimedia training tools for e-learning in Telecommunications and Networking, Automotive electronics.

Kari Kosonen has a background in the developmental psychology. He is currently working as a researcher at the Department of Psychology in the University of Helsinki in an international project developing pedagogical and technological solutions for supporting collaborative knowledge creation processes. Prior to this work Kosonen served as a certified developmental psychologist in a family counselling centre providing diagnostic and rehabitational services for learners with specific pedagogical needs. In his current work and doctoral studies he designs and tests conceptual scaffolding tools for addressing specific learning challenges stemming from the complexity of a target domain or a learner's own individual needs. Kosonen applies and explores a theoretical framework based on the Activity theory and particularly the writings of Vygotsky, Leontiev and Galperin that he was familiarized with, in their original language, during his psychologist training at the Moscow State University.