

Facilitating Shared Knowledge Construction in Collaborative Learning

Stephan Lukosch
 Fern Universität in Hagen
 Department of Mathematics and Computer Science
 E-mail: stephan.lukosch@fernuni-hagen.de,
<http://kalu.fernuni-hagen.de>

Keywords: collaborative learning, shared knowledge construction, game-based learning, web-based learning communities

Received: March 16, 2007

The German distance learning university uses the web-based collaborative learning platform CURE to support different collaborative learning scenarios, e.g. collaborative exercises or virtual seminars. During these scenarios students form learning groups upon teacher's request and collaborate on a common task. But as soon as the given tasks are accomplished, in most times the collaboration stagnates and finally stops. In this paper, we report on extensions to CURE that were designed to foster shared knowledge construction and allow learning in a entertaining way. These extensions were developed in a participatory process with the students. To enable the students to design and develop these extensions, we used patterns for computer-mediated interaction. In our opinion, these entertaining extensions will lead to a higher degree of interaction between the students which leads to a shared knowledge and finally a learning community.

Povzetek: Opisan je dodatek pri platformi CURE za sodelovalno učenje.

1 Introduction

The FernUniversität in Hagen is the German distance learning university. Teaching at the FernUniversität includes different forms of learning: courses, seminars, and different forms of practical problem solving in lab courses. Course material and accompanying individual exercises are sent to distributed students via surface mail or the Internet. As the students are distributed all over Germany, it is difficult for them to find appropriate co-learners and to learn together. As a result, students at the FernUniversität primarily learn individually, feel isolated, lack practice of collaboration, and miss the motivation that teamwork and team members may provide.

A survey including students and different faculties showed a major interest in collaborative learning scenarios and the FernUniversität built the collaborative learning platform CURE [9]. Up to now, students mainly use CURE to form learning groups upon teacher's request. In these groups, they discuss course content, solve assignments, or collaboratively write a seminar thesis. This cooperation and discussion works well as long as there is a group task given by the teachers. If the given tasks are accomplished, the collaboration in most cases stagnates or even finally stops.

In our opinion, a learning community could help to increase collaboration among the students. However, communities cannot be designed. Instead, learning communities evolve through the collective building of shared knowledge and the shifting participation of their members [15] and only the software that supports the

community is designed [22]. There are some key factors for a successful online community. Wenger [27] and Haythornthwaite et al. [11] are of the opinion that participation and practice are the key factors for developing a learning community. Palloff and Pratt [20] emphasize that community members must have possibilities for shared knowledge construction.

CURE can serve as a basis for a learning community if students could be motivated to a higher degree of collaborative interaction and shared knowledge construction. Our approach to reach this goal consists of combining entertaining learning, e.g. based on learning games [23], and participatory design. In our opinion, entertaining learning approaches can be used to increase the motivation for more frequent collaborative interaction which may result in the construction of shared knowledge. Due to a higher degree of collaborative interaction, the students will get to know each other much better and develop responsibilities for their peer students.

Therefore, we extended CURE with learning gadgets, i.e. entertaining tools for collaborative learning. To ensure that these learning gadgets will be accepted by the students, we employed a participatory approach to these new forms of educational interaction. In a lab course, we let our students suggest learning gadgets which in their opinion would foster collaborative interaction and support shared knowledge construction.

To enable our students to design such learning gadgets, we equipped them with a pattern language for

computer-mediated interaction [24]. Patterns can serve as a *Lingua Franca* for design [6] that help end-users and developers in communication and as an educational and communicative vehicle [4; 14].

In the following sections, we first describe CURE in more detail before we focus on the participatory process, the patterns, the resulting learning gadgets, and finally give an outlook on future work.

2 CURE in a nutshell

CURE [9] is a web-based collaboration space. It was developed to support the initial scenarios of collaborative exercises, seminars, lab courses, and the preparation of theses.

Students can structure their interaction in groups that inhabit virtual rooms. Room metaphors [7; 21] have been widely used to structure collaboration. Figure 1 shows the abstractions that are offered by CURE. Users enter the cooperative learning environment via an entry room that is called Hall. Rooms can contain pages, communication channels, e.g. chat, threaded mail, and users. Users, who are in the same room at the same time, can communicate by means of a synchronous communication channel, i.e. by using the chat that is automatically established between all users in the room. They can also access all pages that are contained in the room. Changes of these pages are visible to all members in the room. The concept of a virtual key [10] is used to express access permissions of the key holder on rooms. The access permissions distinguish rights to enter a room, create sub rooms, edit pages, or to communicate within the room. Rooms with public keys are accessible by all registered users of the system.

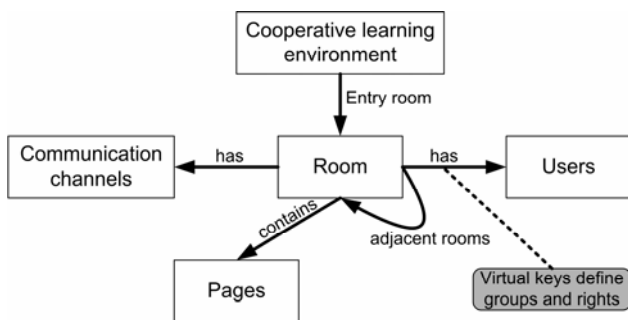


Figure 1: CURE abstractions.

Users can enter a room to access the room’s communication channels and participate in collaborative activities. Users can also create and edit pages in the room. Pages may either be directly edited using a simple Wiki-like syntax [16], or they may contain binary documents or artefacts. In particular, the syntax supports links to other pages, other rooms, external URLs or mail addresses. The server stores all artefacts to support collaborative access. When users leave the room, the content stays there to allow users to come back later and continue their work on the room’s pages.

Figure 2 shows a typical room in CURE. The numbers in the figure refer to details explained in the following paragraphs. A room contains documents (1)

that can be edited by those users, who have sufficient edit rights (2). CURE stores all versions of a page. Users can browse different versions (3) to understand their colleagues' changes. Communication is supported by two room-based communication channels, i.e. a mail box (4) and a persistent chat (5). Users can use the room-based e-mail to send a mail to the room. Users of the room that have sufficient communication rights will receive this message.

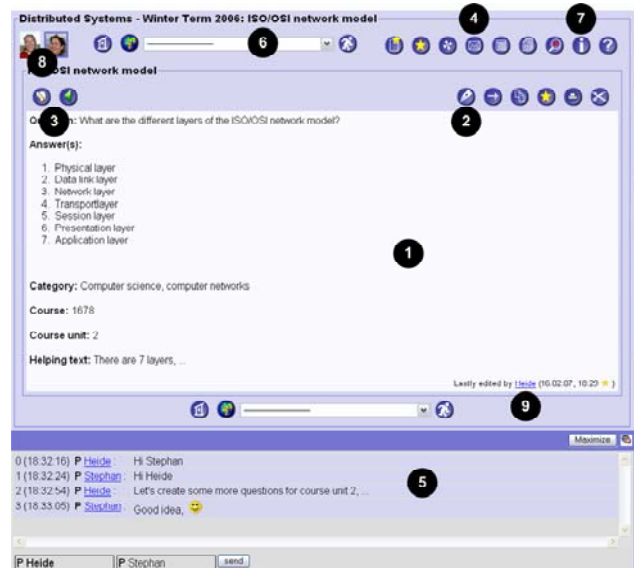


Figure 2: A room in CURE

By providing a plenary room, sharing and communication in a whole class or organization can be supported. By creating new rooms for sub-groups and connecting those to the classes' or organization's room, work and collaboration can be flexibly structured. Starting from the plenary room users can navigate to the connected sub-rooms (6).

For user coordination, CURE supports various types of awareness information:

- Users can see in the room's properties who else has access to this room (7).
- Users can see which users are currently in this room (8).
- If the chat is enabled in the room, users can directly start chatting to each other (5).
- Users can see who has lastly edited the current page (9).
- Daily reports automatically posted to all users of a room include all changes made since the last report was sent.

3 Pattern-based participatory design of the learning gadgets

Each year our department conducts a practical lab course in which groups of up to 8 students collaboratively develop a collaborative application. At the beginning of one of these lab courses, we asked the students to suggest learning gadgets for the CURE environment that

- assist them in collaborative learning,

- help to build a shared knowledge,
- motivate them to become an active member of a learning community, and
- foster learning in an entertaining way.

As starting point for further investigations or thoughts, we suggested game-based learning approaches [23] to the students, as these are well-known to increase the motivation for more frequent collaborative interaction. As result of our basic requirements, the students suggested 23 learning gadgets that in their opinion fulfil our requirements. Most of the suggestions focused on the game-based learning approach. After a discussion among teachers and students, some of the proposed learning gadgets were selected and developed in the lab course.

To enable our students to design and develop such learning gadgets, we equipped them with a pattern language for computer-mediated interaction [24] that evolved in our group over the last years. The idea of patterns originates from Christopher Alexander's work [1; 2] in urban architecture. According to Alexander, 'patterns describe a problem which occurs over and over again and the core of the solution to that problem'. Each pattern includes a problem description, which highlights a set of conflicting forces and a proven solution, which helps to resolve the forces. An interconnected set of patterns is called a pattern language. Patterns of a pattern language are intended to be used together in a specific problem domain for which the pattern language guides the design decisions in the specific problem domain.

Especially, when developing applications that support collaborative interaction, end-user involvement is a crucial issue [17]. To foster communication between developers and end-users, they need a common language and understanding of the problem space to determine the requirements. Pattern languages are an educational and communicative vehicle for reaching this goal. Once the requirements are identified, patterns support developers in implementing the collaborative application by teaching them on how to design and develop groupware applications and reuse proven solutions.

Applications that support collaborative interaction are often considered as *socio-technical systems*, as the technical system has to support the social process of collaboration. Thus, our patterns have to describe the *technology* that supports the *group process* and therefore include a technical and a social aspect.

Compared to software patterns, our patterns need a special form that can be understood by end-users as well as software developers. Our patterns follow the pattern structure outlined in the *Oregon Software Development Process* [25]. A pattern starts with the *pattern name* followed by other possible names for the pattern (*AKA*), the *intent*, and the *context* of the pattern. These first descriptions help readers to decide whether or not the pattern may fit in their current situation.

The problem section of a pattern contains the *problem* statement in bold font, followed by a *scenario* and typical *symptoms*. The *scenario* is a concrete description of a situation where the pattern could be used, which makes the tension of the *problem* statement

(the conflicting forces) tangible. The *symptoms* describe typical observations that indicate the problem exists.

The solution section of the pattern explains the actual *solution* to the problem. The *dynamics* section states the main components or actors that interact in the pattern and explains how they relate to each other. The *rationale* section explains the impact of the solution on the various (conflicting) forces involved. Unfortunately, applying a pattern may introduce new unbalanced forces. These counter forces are described in the section labelled *danger spots*.

The solution presented in a pattern represents a proven solution to a recurring problem, so the *known uses* section provides well-known examples where this pattern is applied. Finally, the *related patterns* section states what patterns are closely related to this one, and with which other patterns this one should be used.

In the following sections, we will describe two of the selected learning gadgets. All learning gadgets can be added to the interaction possibilities of a room in CURE by simply creating a new page. Thus, each user can initiate collaboration. The learning gadget *Fountain of Wisdom* focuses on learning based on a question and answer paradigm. The questions and answers can be either provided by the teacher or they can be defined by the students themselves. In either case, the question and answers are collected in a common question repository which serves as shared learning knowledge. The learning gadget *One for all and all for one* focuses on collaborative exam preparation by giving distributed slide presentations [18]. The slide presentations can be shared in the learning community and thus again help to construct a shared learning knowledge. For space reasons, we will not go into details of the implementation, but give thumbnails, i.e. the problem and solution statement of the pattern, of the patterns (pattern names are set in SMALL CAPS) that can be identified in the learning gadgets.

3.1 One for all and all for one

This learning gadget allows student groups to give distributed slide presentations. These presentations are available to all members of the room in CURE in which the presentations take place. Each room member can join the COLLABORATIVE SESSION at any time by entering the presentation page which offers all functionality for the distributed presentation.

COLLABORATIVE SESSION

Problem: Users need a shared context for synchronous collaboration. Computer-mediated environments are neither concrete nor visible, however. This makes it difficult to define a shared context and thereby plan synchronous collaboration.

Solution: Model the context for synchronous collaboration as a shared session object. Visualize the session state and support users in starting, joining, leaving, and terminating the session. When users join a session, automatically start the necessary collaboration tools.

Before a presentation can be started, one group member has to be selected as presenter. For that purpose, *One for all and all for one* lets group members apply as presenter for a specific topic. Then, the group decides in a VOTE whether the application is accepted or not (see Figure 3) and thereby selects the presenter for a topic.

VOTE

Problem: It is hard to work out the distribution of opinions in the community. However, good understanding of other users' attitudes can be important when making decisions.

Solution: Provide an easy means of setting up and running a poll. Show a virtual ballot in a prominent place in the community. After the vote is over, present the result.

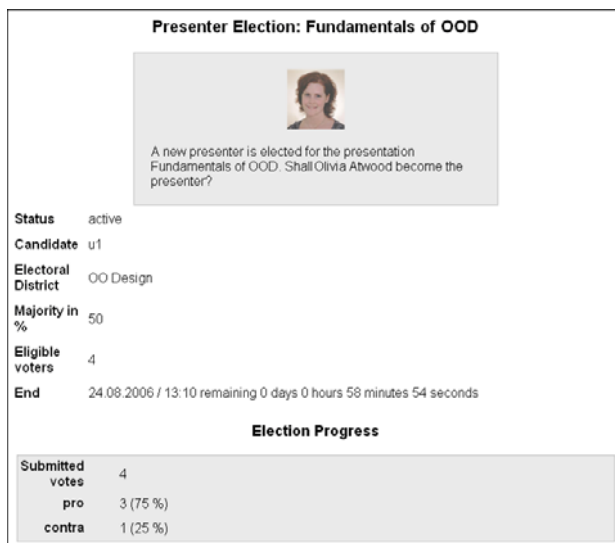


Figure 3: Group VOTE in *One for all and all for one*

In the positive case, the presenter prepares slides and gives a synchronous slide presentation in CURE. For preparing the slides, the presenter can make use of the CURE Wiki syntax. At the beginning of the presentation, the presenter switches the prepared slides to the presentation mode.

Other students who visit the presentation's page will also join the presentation but use a dedicated listener's user interface. This interface includes different means for interacting with the presenter. Most important, it creates a shared focus on a common slide. Whenever the presenter switches the slide, all listeners will automatically follow the presenter, thereby implementing the SHARED BROWSING pattern.

SHARED BROWSING

Problem: Users have problems finding relevant information in a collaboration space. They often get lost.

Solution: Browse through the information space together. Provide a means for communication, and collaborative browsers that show the same information at each client's site.

One for all and all for one makes use of the EMBEDDED CHAT in CURE which allows the audience to communicate with each other or with the presenter (see bottom of Figure 4).

EMBEDDED CHAT

Problem: Users need to communicate. They are used to sending electronic mail. But since e-mail is asynchronous by nature, it is often too slow to resolve issues that arise in synchronous collaboration.

Solution: Integrate a tool for quick synchronous interaction into your cooperative application. Let users send short text messages, distribute these messages to all other group members immediately, and display these messages at each group member's site.

Users can decide whether they post a message to the EMBEDDED CHAT or send the message directly to the presenter to ask a question. Questions for the presenter are collected at the presenter's view and can be answered by the presenter as soon as it fits into the presentation. Thereby, *One for all and all for one* implements the FEEDBACK LOOP pattern.

FEEDBACK LOOP

Problem: In any communication, the recipient of a message can only refer to the message in order to understand it. However, most messages are ambiguous.

Solution: Provide an easy means for readers to contact the author. Create a user interface element close to the content that opens input fields for the reader's questions and feedback.

To give the presenter even more feedback, group members can indicate their current degree of understanding in a barometer of opinion. The individual scores for each student are shown in the USER LIST on the left side of the screenshot in Figure 4.

USER LIST

Problem: Users do not know with whom they are interacting or could interact. Consequently, they do not have a feeling of participating in a group.

Solution: Provide awareness in context. Show who is currently accessing an artifact or participating in a COLLABORATIVE SESSION. Ensure that the information is always valid.

The average understanding of the audience is shown to all users in the upper part of the presentation view (see top of Figure 4). The challenge for the presenter is to keep all members of the audience on a high level of understanding, as it is the case for Olivia Atwood in Figure 4.

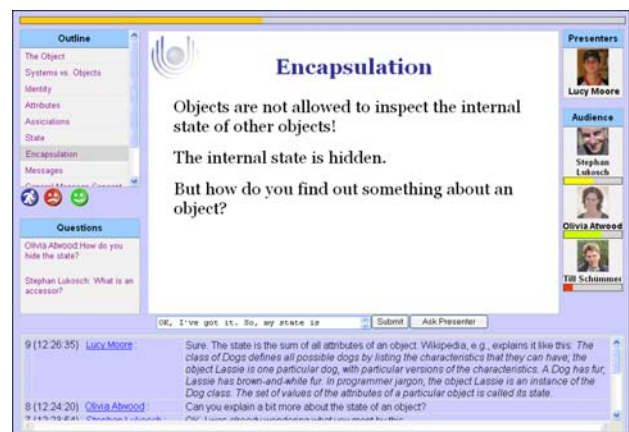


Figure 4: Presentation in *One for all and all for one*

The average understanding is tracked during the whole presentation and will be displayed together with the slides after the presentation is over (see Figure 5).



Figure 5: Average understanding in *One for all and all for one*

Once the presentation is over, the presenter can release the slides for asynchronous access which allows the users to either review the slides only or review the whole COLLABORATIVE SESSION. In the latter case, *One for all and all for one* implements the Persistent Session pattern and not only shows the slides but also the questions and the chat log.

PERSISTENT SESSION

Problem: After interacting in a COLLABORATIVE SESSION, users want to resume their collaboration with the results achieved, or want to review them, but the results are not available.

Solution: Persistently store the results of a synchronous COLLABORATIVE SESSION on a central server. Keep a master copy of the shared data and track all changes that are applied to it. Let users access the master copy at the central server for review or session resumption purposes.

In summary, *One for all and all for one*

- assists students in learning, as it allows students to collaboratively discuss learning material,
- facilitates shared knowledge construction, as the slides together with the students' feedback can be shared in the learning community and can be used by other students as learning material,
- motivates students to become an active member of the community by giving direct feedback to the presenter, and
- fosters learning in an entertaining way by challenging the presenter to create good feedback curves and the students in the audience to have an understanding that is better than the rest of the audience.

3.2 Fountain of Wisdom

Fountain of Wisdom is based on a 3D virtual maze in which two teams compete with each other by answering questions. Additional teams can play in parallel in the same maze. Users can meet on a so-called 3D marketplace and use the EMBEDDED CHAT to socialize with co-learners and propose a game on a specific topic.

Figure 6 shows a screenshot of the marketplace. On the left you can see a list of users that are currently on the marketplace. The bottom of the screenshot shows the chat. In the 3D view, you can see some one additional player, i.e. the snowman, and that the local user is currently proposing a game. The local user has chosen the topic 1678, which is the number for a course on distributed systems, and is going to send an INVITATION to the user Stephan which asks him to participate in the COLLABORATIVE SESSION.

INVITATION

Problem: One user wants to interact with another. The other user may be unavailable or busy in another context so that an immediate collaboration would disturb them.

Solution: Send and track invitations to the intended participants. Include meta-information on the intended COLLABORATIVE SESSION. Automatically add all users who accept the invitation to the COLLABORATIVE SESSION.

Finally, the local user has chosen to limit the duration of the game to 10 minutes. Another opportunity would have been to limit the game to a number of questions. At the end, the team that correctly answered the most questions wins the game.

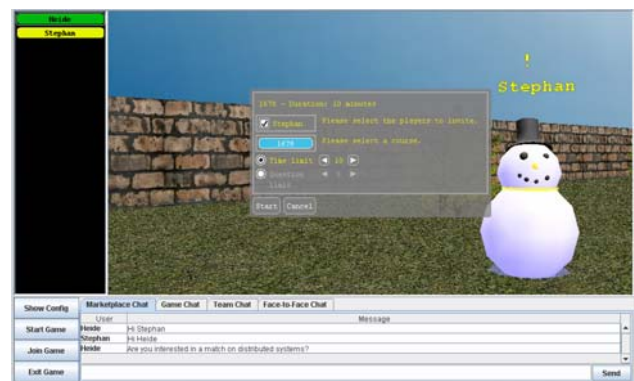


Figure 6: *Fountain of Wisdom* marketplace

After the teams formed, the 3D game maze is initiated. Users that later on visit the marketplace can join the game and decide in which of the two possible teams they want to participate. The underlying system then performs a STATE TRANSFER.

STATE TRANSFER

Problem: Users are collaborating in a COLLABORATIVE SESSION but not all of them participate from the beginning. Due to this, some do not know the intermediate results of the COLLABORATIVE SESSION which makes it difficult for them to collaborate.

Solution: Transmit the current state of shared objects to latecomers when they join a COLLABORATIVE SESSION. Since

all current participants have the most recent state of the session's shared objects, the system can ask any of the existing clients to perform the state transfer. Ensure the consistency of the state.

The maze provides fountains from which team members have to obtain a question. Complementary to the fountains, there are several sinks in the maze where possible answers to the questions can be found. Each sink contains answers to a number of questions. When users step onto a sink, their view changes and displays the answers to all questions associated with the sink. From this list of possible answers the user has to choose the correct ones knowing only the number of correct answers. Each correct answer counts for individual and team points. Figure 7 shows the view for local users when they step onto a sink. The current question is shown in the header of the 3D view. Left to the 3D view, you can see the USER LIST, which highlights the teams competing with each other. The bottom of the screenshot again shows the EMBEDDED CHAT.

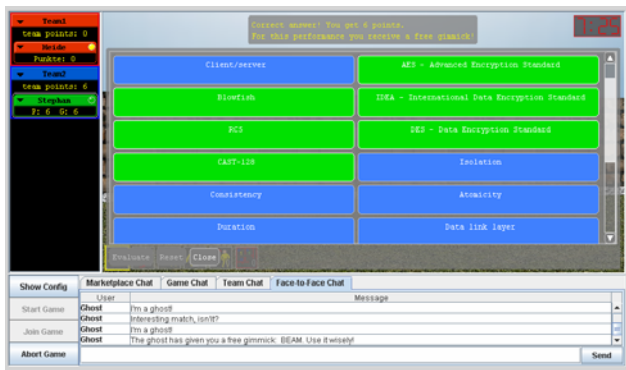


Figure 7: Fountain of Wisdom sink view

Figure 8 shows the normal maze view of *Fountain of Wisdom*. From the view of the local user, you can see another user, i.e. the snowman on the right side, and one bad ghost on the left side. The upper right corner of the 3D view shows a small ACTIVE MAP of the maze with the positions of the other users in the maze and the time that is left in this match.

ACTIVE MAP

Problem: To orient themselves and interact in space, users have to create a mental model that represents the space and the artifacts and users it contains. This is a difficult task.

Solution: Create a reduced visual representation of the spatial domain mode lby means of a map. Show other users' locations on the map. Ensure that the map is dynamic for artifacts and users, but static with respect to landmarks.

At the bottom of the 3D view you can see different chats, i.e. one for the marketplace, one for all participants of the current game, one for the team, and one for the user next to oneself. The chats can be used to gain information about where to find the correct sink for a question and to discuss possible answers with team members.

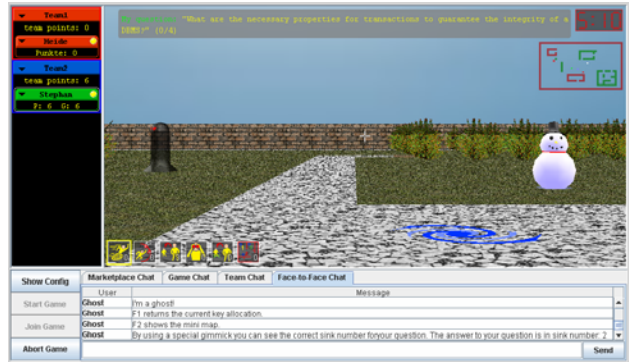


Figure 8: Fountain of Wisdom maze view

Apart from bad ghosts there are also good ghosts in the maze. Good ghosts help the players by e.g. giving tips on the correct answer to a question or giving away gimmicks that allow the user to perform special actions, e.g. to move faster or beam from one place to another in the maze. If a player comes too close to a bad ghost, this may steal the current question and the player has to get a new one.

Prior to a game, students can define questions and answers for a specific topic as CURE pages. CURE supports so-called Wiki-templates [8], which allow end-users to define form-based pages. These Wiki-templates can be used to structure shared knowledge and simplify its construction. Figure 9 shows a screenshot of a typical question and answer page in CURE. For all question and answer pages the same Wiki-template is used to enable re-use in different learning gadgets. Apart from the question, such a page consists of a number of correct answers, a question category, the specification of the course, and a helping text that is used by the good ghosts to provide hints about the correct answer of the question.

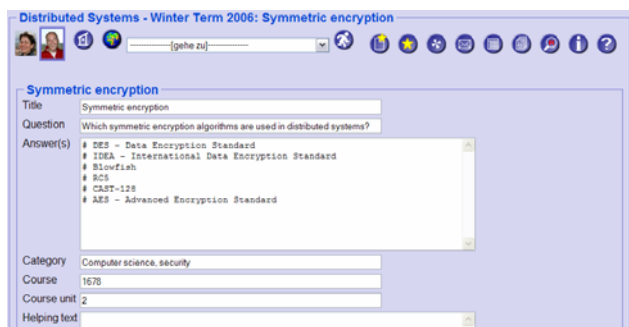


Figure 9: Question and answer page in CURE

The question and answer pages are elements of a shared question repository in CURE and define the topics that are available for a game. To ensure the quality of the questions, CURE, implements a combination of the patterns QUALITY INSPECTION and VOTE.

QUALITY INSPECTION

Problem: Members participate in a community to enjoy high-quality contributions from fellow members. However, not every contribution has the same quality. Low-quality contributions can annoy community members and distract their attention from high-quality gems.

Solution: *Select users as moderators and let them release only relevant contributions into the community's interaction space. Give moderators the right to remove any contribution and to expel users from the community.*

Due to the implementation of the patterns QUALITY INSPECTION and VOTE, students can rate the quality of a question and the corresponding answers using a special page CURE provides (see Figure 9). The ratings of all students are accumulated and shown as stars in the last column of the table in Figure 9. Additionally, students can act as quality inspectors by removing questions from the repository that are of low quality.

Course	Course unit	Question	Rating
1678	2	What kinds of group awarene...	☆☆☆
1678	2	What asymmetric encryption ...	☆☆☆
1678	2	What are the different laye...	☆☆☆
1678	2	What kind of network archit...	☆☆☆
1678	2	What are the necessary prop...	☆☆☆

Figure 9: Rating in the shared question repository

Summarizing, *Fountain of Wisdom*

- increases social interaction, as students can meet on the marketplace to form teams and during a game the different chats allow cross-group interaction,
- strengthens the group feeling, as the students cooperate in groups to gain as much group points as possible, and
- allows students to self-organize their learning and their interaction, as each user can create rooms in CURE and define the interaction possibilities of a room, and
- supports collaborative learning by constructing a shared question repository that can be used by all students of the CURE environment.

4 Related Work

Most web-based learning platforms focus more on the distribution of learning material than social interaction or possibilities to construct shared knowledge. *Blackboard* [3] offers a virtual classroom, in which interaction can take place. Students and teachers can collaborate on a shared whiteboard and communicate via chat. To support peer collaboration teachers can offer group projects in which each group can be given its own file exchange area, discussion board, etc. But the results of these group projects cannot become part of a shared knowledge.

Centra Live for Virtual Classes [5] organizes collaborative learning activities as events. Students can enrol in events by browsing a catalogue of upcoming events. For these events, teachers can plan a variety of synchronous interaction using the provided functionality, e.g. polls, surveys, chat, whiteboards, cooperative Web browsing, etc. Though these tools might foster interaction, the collaboration is not self-initiated and does not focus on the construction of a shared knowledge.

Moodle [19] offers a lot of activity modules that can be associated with learning material. Among these activity modules are, e.g., a Wiki, a glossary, or a quiz. The quiz mainly focuses on assessment. The glossary and the Wiki allow interaction and collaboration among the students but must be enabled by the teacher. As the teacher has to enable all these activity modules self-initiated interaction and collaboration among the students is not possible.

BSCW [26] is based on *BSCW* and offers a group of students a web-based shared workspace. Compared to other learning platforms, *BSCW* offers special tools for collaborative knowledge building, but again, all interaction and collaboration possibilities are defined by the teacher.

ILIAS [13] or *KOLUMBUS* [12] are further learning environments that offer similar functionalities, but like the above platforms do not support entertaining interaction.

Compared to other learning environments, CURE in combination with the described learning gadgets offers a much higher degree of collaborative interaction and participation possibilities. In our opinion, this provides a good means to motivate students for continuous participation in a learning community. The question repository and the shared slide presentation allow students to construct a persistent shared knowledge which might serve as glue for forming a learning community. Students will not feel isolated anymore as they on the one hand can benefit from the existing repository and on the other hand can become an active member of the learning community by participating in the construction of the shared repository.

5 Conclusion

Up to know there is no long-time study of CURE in combination with the learning gadgets. At the end of the lab course, we asked our students if they would use the learning gadgets for collaborative learning. Almost all students indicated their interest. Additionally, the participatory design of the learning gadgets, which is based on the re-use of proven solutions in the form of patterns for computer-mediated interaction [24], suggests that the learning gadgets will be accepted and used by the students. For that reason, we are currently planning long-time studies of the learning gadgets and add the learning gadgets to the deployed version of CURE. We plan to evaluate the usage and impact of the learning gadgets on how they foster collaborative interaction, shared knowledge construction, and the building of a learning community.

Acknowledgement

Special thanks are due to Mohamed Bourimi and Till Schümmer for their engagement in supervising two of the lab groups and to all members of the lab groups (in alphabetical order): Yves Albrecht, Lukas Beyer, Marco Blum, Christian Brandtner, Kathrin Dentler, Volker Engels, Thomas Grasse, Matthias Hellweg, Knut Linke,

Irini Ntokoutsi, Frank Plieninger, Martin Rasel, Franz Schinerl, and Julia Schmeisser.

References

- [1] Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I., Angel, S., (1977). A pattern language. Oxford University Press, New York.
- [2] Alexander, C., (1979). The timeless way of building. Oxford University Press, New York.
- [3] Blackboard Inc. (2006). Product homepage. <http://www.blackboard.com/>, last visited March 2007.
- [4] Brugali, D., Menga, G., Aarsten, A. (1997). The framework life span. *Communications of the ACM* 40 (10), pages 65–68.
- [5] Centra Live Suite (2007). Product homepage. <http://www.saba.com/products/centra/live/>, last visited March 2007.
- [6] Erickson, T. (2000). Lingua francas for design: sacred places and pattern languages. In: *Proceedings of the conference on Designing interactive systems*. ACM Press, pages 357–368.
- [7] Greenberg, Saul and Mark Roseman (2003). Using a Room Metaphor to Ease Transitions in Groupware. In M. Ackermann, V. Pipek, and V. Wulf, editors, *Sharing Expertise: Beyond Knowledge Management*, pages 203–256. MIT Press, Cambridge, MA, USA.
- [8] Haake, Anja, Stephan Lukosch, and Till Schümmer (2005). Wiki-Templates: Adding Structure Support to Wikis On Demand. In: *WikiSym 2005 – Conference Proceedings of the 2005 International Symposium on Wikis*, Seiten 41-51. ACM Press.
- [9] Haake, Jörg. M, Till Schümmer, Anja Haake, Mohamed Bourimi, and Britta Landgraf (2004). Supporting flexible collaborative distance learning in the CURE platform. In *Proceedings of the Hawaii International Conference On System Sciences (HICSS-37)*. IEEE Press.
- [10] Haake, Jörg M., Anja Haake, Till Schümmer, Mohamed Bourimi, and Britta Landgraf (2004). End-User Controlled Group Formation and Access Rights Management in a Shared Workspace System. In *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 554–563, ACM Press.
- [11] Haythornthwaite, Caroline, Michelle M. Kazmer, Jennifer Robins, and Susan Shoemaker (2000). Community Development Among Distance Learners: Temporal and Technological Dimensions. *Journal of Computer-Mediated Communication*, 6(1).
- [12] Herrmann, Thomas and Andrea Kienle (2003). KOLUMBUS: Context-oriented communication support in a collaborative learning environment. *Informatics and the Digital Society. Social, Ethical, and Cognitive Issues*, Kluwer, pages 251-260.
- [13] ILIAS (2007). ILIAS open source. <http://www.ilias.de/>, last visited March 2007.
- [14] Johnson, R. E. (1997). Frameworks = (components + patterns). *Communications of the ACM* 40 (10), 39–42
- [15] Lave, J. and Wenger, E. (1991). *Situated Learning Legitimate Peripheral Participation* Cambridge University Press.
- [16] Leuf, Bo and Ward Cunningham (2001). *The WIKI way*. Addison-Wesley, Boston, MA, USA.
- [17] Lukosch, Stephan and Till Schümmer (2006). Groupware Development Support with Technology Patterns. *International Journal of Human Computer Studies*, Special Issue on 'Theoretical and Empirical Advances in Groupware Research', 64(7):599-610.
- [18] Lukosch, Stephan and Till Schümmer (2006): Making exam preparation an enjoyable experience. *International Journal of Interactive Technology and Smart Education*, Special Issue on 'Computer Game-based Learning', 3(4):259-274.
- [19] Moodle, (2007). Moodle - A Free, Open Source Course Management System for Online Learning. <http://moodle.org/>, last visited March 2007.
- [20] Palloff, Rena M. and Keith Pratt (1999). *Building Learning Communities in Cyberspace – Effective Strategies for the Online Classroom*. Jossey Bass Wiley.
- [21] Pfister, Hans-Rüdiger, Christian Schuckmann, Jennifer Beck-Wilson, and Martin Wessner (1998). The Metaphor of Virtual Rooms in the Cooperative Learning Environment CLear. In *Cooperative Buildings - Integrating Information, Organization and Architecture. Proceedings of CoBuild'98*, LNCS 1370, pages 107–113, Springer-Verlag Berlin Heidelberg.
- [22] Preece, Jenny (2000). *Online Communities*. Wiley, Chichester, UK.
- [23] Prensky, Marc (2001). *Digital Game-Based Learning*. McGraw-Hill Education.
- [24] Schümmer, Till and Stephan Lukosch (2007). Patterns for Computer-Mediated Interaction. John Wiley and Sons, Chichester, UK.
- [25] Schümmer, Till, Stephan Lukosch, and Robert Slagter (2006). Using Patterns to empower End-users The Oregon Software Development Process for Groupware. *International Journal of Cooperative Information Systems, Special Issue on '11th International Workshop on Groupware (CRIWG'05)'*, 15(2):259-288.
- [26] Stahl, Gerry (2002). Groupware Goes to School. In *Groupware: Design, Implementation, and Use, 8th International Workshop, CRIWG 2002*, LNCS 2440, pages 7-24, Springer-Verlag Berlin Heidelberg.
- [27] Wenger, Etienne, (1998). *Communities of Practice: Learning, Meaning, and Identity*. Cambridge University Press.