Journal of Universal Computer Science, vol. 15, no. 9 (2009), 1833-1851
submitted: 15/8/08, accepted: 25/4/09, appeared: 1/5/09 © J.UCS

MC-Supporter: Flexible Mobile Computing Supporting Learning though Social Interactions

Nelson Baloian
(Department of Computer Science, Universidad de Chile, Santiago, Chile
nbaloian@dcc.uchile.cl)

Gustavo Zurita
(Management Control and Information Systems Department, Universidad de Chile
Santiago, Chile
gzurita@fen.uchile.cl)

Abstract: Nowadays, we are experiencing a rapid development in mobile computing because of the sinking prices of the mobile devices and the availability of wireless networks that can connect them. The ability of many of these devices to set up ad-hoc networks by proximity allows face-to-face interaction combined with mobility. However, mobile devices are much weaker in computing power compared with desktop or laptop computing. Therefore, a key aspect to ensure success of an application supporting mobile learning is whether mobility is really needed for the activity it supports and if mobile devices do really represent an added value compared with the same application implemented on non-mobile devices. This work presents MCI-Supporter, an application supporting collaborative learning practises in the classroom. MCI-Supported was conceived by first analyzing the best known collaborative learning practices trying to find out which are the real needs for mobility and face-to-face interaction and then designing the application supporting learning activities where mobile computing does really represents an added value compared to the desktop computing scenario.

Keywords: Computer Supported Collaborative Llearning (CSCL), Mobile Learning, Social Interactions
Categories: L.6.2, L.7.0

1 Introduction

Many authors have already stressed the importance of social interactions in the learning process. For example, Vygotsky's sociocultural theory [Vygotsky, 1978] points out at the importance of using artefacts that allow learners to interact promoting knowledge acquisition. Bellamy [Bellamy, 1996] also proposes that learners should create and share artefacts within their community. He also says that educational environments should be designed in order to involve a close collaboration among their peers as well as between students and experts. Many other authors like [Barab, 2000], [Brown, 1989] and [Resnick, 1987] emphasize the social nature of cognition and learning. Lave [Lave, 1988; Lave, 1991], has shown using anthropological approaches that meaning and identity are constructed from social interactions. Wenger states in his Social Theory of Learning [Wenger, 1998] that people learn through participation in activities and that knowledge can be considered as our ability to contribute to valued practices. Participating in a social unit provides

meaning to experiences and activity, and provides shared perspectives and resources for sustaining engagement in activity. Thus the social nature of experience provides motivation for engagement, leads to joint enterprise, and shapes what is learned [SCRG, 2005]. Computer Supported Collaborative Learning (CSCL) is a discipline that promotes the development of computer-based systems facilitating learners' communication, enabling high levels of social interaction in the classrooms [Liu. 2003].

With the popularization of mobile computing devices many authors have highlighted the interesting role that mobile computing can play in the development of CSCL systems fostering social interactions in collaborative learning environments [Liu, 2002] [Zurita., 2008]. Because of their reduced size, mobile devices allow a better face-to-face contact between learners and mobility facilitates the dynamic configuration of the groups during a computer supported learning activity as compared with the situation of learning supported by desktop computing devices.

However, we think that in many cases the use of mobile devices to implement CSCL activities is not fully justified. In fact, mobile devices have many drawbacks compared with desktop computers: they have less computing power, a much smaller screen, and shorter battery life, just to mention a few. For this reason, we think that before implementing a mobile computing-based CSCL system it is important to analyse whether the advantages of mobility compensate the drawbacks that mobile computing imposes for the particular learning activity the system is going to support.

In this paper, we present a prototype of a system called Mobile Collaborative Interaction Supporter (MCI-Supporter) based on PDAs and Tablet-PCs wirelessly interconnected, implementing artefacts and techniques that can improve teacher-student social interaction. In order to design the system's requirements we start by analysing which pedagogical practices can benefit from the mobile situation PDAs can provide despite of their drawbacks.

MCI-Supporter supports various pedagogical practices requiring rich teacher-students and student-student interactions. It helps a teacher to create and distribute two different kinds of problems to the students and provides tools to asses their work when solving them, thus allowing the teacher to provide immediate feedback and promoting reflection. All this is made in real time while allowing at the same time rich face-to-face interaction during the teaching/learning session [Wright, 1995], [Skinner, 1993]. The system makes use of the PDAs' capabilities to automatically build an ad-hoc network among them by proximity implementing a full peer-to-peer architected. This allows the system to be used anywhere, without the presence of neither an external wireless network nor a central server. The next chapter presents related work about CSCL and mobile learning. The third chapter presents various learning practices and the way mobile computing in general and MCI-Supporter in particular is able to improve them. The fourth chapter describes the functionalities that MCI-Supporter implements in order to support the described learning activities. Chapter 5 presents its architecture. Chapter 6 presents a testing experience using the system in a real teaching/learning scenario and chapter 7 concludes the paper.

2 Related Work

Some authors have already pointed out to the fact that research and technology are moving towards ubiquitous computing [Sturm, 2005], where computers are there ready to be used anytime, anywhere, not being the center of attention of the users but integrating swiftly and subtle into the working scenario. Because of its characteristics, mobile devices are very

suitable to integrate these kinds of scenarios. The classroom situation has been also considered by many authors as an interesting scenario for implementing the principles of ubiquitous computing. This term has also been associated with the concepts of pervasive computing and calm technology [Roberts, 2006], and the "electronic classroom" has become a quite popular term to describe a computer-integrated classroom [Müldner, 1996] [Baloian, 2008].

The contribution that mobile computing can offer to enrich the teaching/learning process inside and outside the classroom has been acknowledged soon after the popularization of mobile devices [Rochelle, 2003], and many systems supporting collaborative, mobile learning have been developed since then.

In [Marcelino, 2007] the authors describe an authoring-tool for handhelds supporting educational simulation and modeling, called Sim-H (SIMulation for Handhelds). The software consists of several modules, each one relating to a type of simulation application that can be used in an educational context. In [Bravo, 2005] the authors present an approach to the classroom context by identification process using RFID technology as an implicit input to the system. The main goal is to acquire natural interaction, because the only requirement for the user (teacher or student) is to carry a device (smart label), identifying and obtaining context services. In [Liu. 2003] and [Liu. 2002] we can also find more example of handhelds supporting learning inside the classroom. In [Molina, 2005] the authors argue that mobile devices can make some collaborative learning activities more realistic and accessible using mobile devices because they allow students to benefit from the mobility features of this kind of devices. They also present a system called DomoSim-Mob which supports the learning of designing automated control facilities in buildings and housing design using PDAs.

Although the previous works represent a valuable contribution to mobile computing supporting collaborative learning, we think there is still room for improvement in many aspects: first, the previously developed scenarios use a centralized communication system attached to a non-mobile server, which does not allow the system to be used in settings where a server and a wireless network connecting the handheld devices with the server is not available. Second, the system is designed to support a specific and very structured learning activity. It does not allow to define new problems to be delivered to the students "on the fly" during the learning activity. Third, they offer limited options to assess the students' work "online" while they are working, before they complete the task. Nor they offer flexible mechanisms to reconfigure the working groups during the learning activity.

Reconfiguring groups dynamically during the students' work according to their performance is a well known collaborative learning best practice [Johnson, 1996]. According to the available literature, this feature has not been supported yet by any existing system supporting mobile learning in the classroom.

3 Pedagogical practices supported by mobile computing

Suthters, [Suthers, 2005] holds that CSCL should focus on the design of technologies that facilitate and foster teacher-students and student-student interactions, as this is a requirement for interactional and especially intersubjective learning epistemologies. An intersubjective learning epistemology goes beyond an information sharing conception of collaborative learning in two ways: 1) it can be about sharing interpretations as well, and 2) these interpretations can be jointly built through social interaction among participants, in addition

to those created by individuals. Intersubjectivity is to be understood in a participatory sense, and may involve disagreement. In this epistemology, learning is not only accomplished through the interactions of the participants, but learning also consists of those interactions themselves [Koshmann, 2005].

Developmental learning through social interaction can be understood as the internalization of interpersonal processes as intrapersonal processes, [Vygotsky, 1978]. Therefore, the new practices in CSCL must be reflected in concomitant creation of novel technological artefacts that support and help to replicate these practices of internalization.

3.1 Social Interaction in CSCL

Active participation of individuals conforming a learning group, as well as dynamic construction of meanings by higher order cognitive processes is required in order to achieve high levels of social interaction. This may involve logical thinking, conceptualizing, analyzing, reasoning, and evaluating [Anton, 1999]. Based on this point of view, construction of meanings and knowledge during social interactions involves making students express their ideas and respond to others'. According to [Anton, 1999], some critical factors of promoting social interaction in the classroom context are: 1) Motivating students' participation: the teacher can trigger the learner's disequilibrium through posing just-in-time appropriate and meaningful open problems. This will motivate them to inquire, and enhance the need of collaboration by assessment and feedback. 2) Focusing students' attention: to ensure that students retain their focus is an important task in classroom interaction. For this, the teacher can provide coaching to the student [Baloian, 2008]. 3) Externalizing internal thinking: thinking does not "expresses" itself but must be shared by creating and exchanging artefacts reflecting ideas, for example during reflection, and challenge-based learning [Baloian, 2006]. 4) Constructing mutual and collaborative understanding: the interlocutors' intention changes constantly before, during, and after the process of interaction. This affects how he/she expresses thinking and responds to the others (peer-review).

It is difficult to get students develop collaborative strategies in the classroom setting [Collazos, 2003]. Collaborative knowledge building in a classroom often requires students' mobility in order to establish face-to-face interactions. Mobile CSCL (MCSCL) applications enable three types of interactions among members in the classroom [Lagos, 2007], including: (1) one-to-one interaction between two students either in the same or in different groups; and between a student and the teacher (2) one-to-many communication between the instructor and students; and (3) many-to-many communication among students. Nevertheless, mobile CSCL design cannot effectively improve students' social interaction (and learning) without the support of appropriate pedagogical practices embedded on the design of the application.

With MCI-Supporter we do not want to propose a new pedagogical practice and test its effectiveness but take the existing ones recognized as best practices and support them in order to enhance the social interactions. From all the practices described by Barkley [Barkley, 2005] as "best practices" for collaborative learning, we now describe which we think are those most suitable to be supported by a mobile collaborative application and how our system supports them:

Problem based learning: The characteristics of this learning practice are: 1) learning is driven by open-ended problems; 2) students work individually or in small collaborative groups; 3) teachers take on the role of "facilitators". Accordingly, students are encouraged to

take responsibility for themselves and the group, organizing and directing the learning process with support from a teacher. It can be used to enhance content knowledge and foster the development of communication, social interaction, problem-solving, and collaborative as well self-directed learning skills.

Using a mobile computing device, the teacher sends to the students previously created open problems or problems created "on the fly" during the learning session. This material is distributed in real time and can be set either to individuals, to a certain students group or to the whole class. While sending this material, the teacher can still interact face-to-face with the students in order to explain, contextualize and/or clarify the problem the students have to work on.

Assessment: It is a participatory, iterative process that: 1) provides information required by the teacher in order to improve teaching and learning; 2) produces evidence about the learning outcome of the students; 3) evaluates whether changes made improve/impact student learning, and documents the learning and teacher's efforts. Assessment of in-class students' work is a fundamental activity, since only when teachers are able to determine the learning process and the learning stand of the students they are also able to provide the students with the right learning environment (learning material plus learning activities) at the right moment [Grüntgens, 2004]. This is also coherent with Cotton, who says "The body of educational research literature which has come to be known as the effective schooling research identifies the practice of monitoring student learning as an essential component of high-quality education" [Cotton, 1988].

With MCI-Supporter the teacher can monitor each student individually or the whole group displaying on his device the students' workspace. The teacher can physically approach a student or a group in order to give feedback in a face-to-face manner. Additionally the teacher can use the mobile device to join the workspace of the student or the group at the same time.

Coaching: It is a learning technique that involves observing individual or collaborative work and providing advice to enhance performance or correct deficiencies from "behind the scene". It used: 1) to develop or provide new skills through on-the-student training; 2) to set learning objectives and expectations together with the individual or among the group participants; 3) to mutually develop and agree on a course of action for enhancing performance; 4) to facilitate learning and enhance performance through using observation, listening, and guidance skills. Give constructive advice and encourage and reward accomplishments

With MCI-Supporter the teacher can join the workspace of an individual student or a group of them working together. In this way, the teacher can add annotations and work together leaving hints for the solution if necessary, complementing this device mediated communication with face-to-face explanations

Reflection: It allows students to take a metacognitive stance to their involvement in the project to explore their own individual and collaborative learning and to determine how the experience has increased their abilities for future academic and professional experiences, as well as informed them as to what skills they need to strengthen. Effective learning situations require time for thinking. Students also reflect on themselves as learners when they evaluate

the thinking processes they used to determine which strategies worked best. They can then apply that information about how they learn as they approach learning in the future.

Feedback: It is used to inform learners about the quality and/or accuracy of their responses. Feedback is distinguishable according to its content, which is identifiable by: 1) load, i.e. the amount of information given from yes-no statements to fuller explanations; 2) form, i.e., the structural similarity between information in the feedback compared to that in the instructional presentation; and 3) type of information, i.e., whether the feedback restated information from the original task, referred to information given elsewhere in the instruction, or provided new information. Researchers recommended immediate feedback for conventional educational settings. Some types or levels of feedback include: 1) a mark or grade or success/fail classification of outcome; 2) the right answer; 3) procedural or surface explanation of the right answer; 4) explanation of what makes the right answer correct: of why it is the right answer; and 5) explanation of what's wrong about the learner's answer.

Challenge-based Learning: In CBL the question or the problem proposed is replaced by a challenge, which the student has to face recurring to different resources that might be available. The assignments or "challenges" to be solved might include ways to develop, design and implement solutions for problems related to scientific phenomena. A meaningful learning activity consistent with CBL is to present learners with a challenge scenario and to ask them to think about a number of possible solutions using a variety of interactive tools. Such an activity serves to centre thinking on meaningful problems and is typically effective in facilitating small group collaboration.

MCI-Supporter allows the teacher to generate and send open questions or "challenges" to the students (individuals or group) during the lecture in three different ways: 1) the teacher can send not fully delimited problems or questions, created on the fly specified by incomplete free-hand drown sketches generated and edited by gestures. 2) same as 1 but the teacher can send along a selection option set to chose the correct answer, 3) same as 1 but with multiple choice option, or 4) same as 1 but sending a sequence of selection (selection of answers in the right sequence). Correspondingly, the answers from the students can be totally open, based on sketches or closed, based on selection. Teacher can use face-to-face interaction to discuss the questions and answers with the students.

The interface allows the teacher to create and send material to the students to work on without having to interact face-to-face. Since the material can be free hand created sketches, the teacher can modify them or add new elements to during the learning session thus the original problem can "evolve" according to the performance of the students.

Mobile Collaborative Learning (MCSCL): Mobile technology opens up potentials for students to work collaboratively while they are in movement, rather than working with allocated partners at a fixed desktop. Students can move inside the classroom and interact with other students in any way that they need. In a MCSCL application it is possible to recognize the technological and the social network: they can communicate face-to-face or by means of the interconnected mobile computing devices.

With MCI-Supporter the teacher can reconfigure the working groups during the exercise in real time, as well as send new problems in order to put students to work together which best fit together taking in account their background knowledge, working style, personal characteristics, etc. Students belonging to a same group can open a shared workspace and work synchronously. Group reconfiguration is also necessary when implementing the Jigsaw

learning practice (Aaronson, 1997) with mobile computer support. Especially during group reconfiguration face-to-face communication between students and teacher and among students themselves may be of crucial importance to coordinate the new situation

Peer-Assisted Leaning: Peer-Assisted Learning (PAL) involves students consciously assisting others to learn, and in so doing, learning more effectively themselves. PAL encompasses peer tutoring, peer modelling, peer education, peer counselling, peer monitoring, and peer assessment, which are differentiated from other more general "cooperative learning" methods.

With MCI-Supporter the teacher can select a certain workspace of a student or group of students in order to present it to the rest of the class if there is an "interesting solution" which may be a good, or a novel solution or a "near miss". At the same time, the teacher uses face-to-face communication to explain the interesting characteristics of the solution being shown.

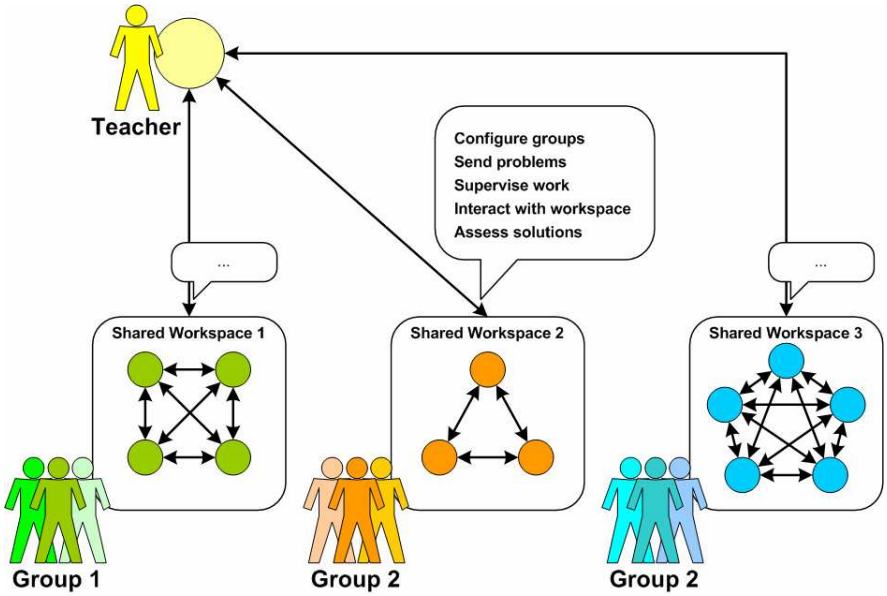

Figure 2: Interactions between teacher's and students' modules

4 Description of MCI Supporter

The MCI-Supporter has been developed to address the pedagogical practices and interactions described in the previous section. For this, the teacher must be able to define groups, assign students to them, create and send problems and asses the students' work by visualizing and joining their workspaces. The students should be able to work on their assigned problems collaboratively and send the answers to the teacher. An application implementing a teacher's and a student's module implementing these actions was developed. The application runs either on PDAs and Tablet-PCs, automatically adapting the interface in order to fit to the respective screen. It has a teacher's and a students' module. The figure 2

shows the interactions between the teacher's and the students' module, and the interaction between the students' modules among them.

Groups' setup: In order to implement collaborative learning, working groups have to be established. This has to be done by the teacher in order to have groups with equal number of participants and distribute them according to their skills. Thus the teacher module has to implement this action. In order to configure the groups, the teacher must enter the "Group making" mode. By clicking the "Create group" icon she will create a new "Group sandbox". A group sandbox will represent the common workspace of a group. Each sandbox displays a scaled version of the group's workspace in real time. For every student participating in the activity an icon with her/his name will be displayed. Several sandboxes are shown in the teacher device, allowing her/him to have an overall view of what all groups are doing. Figure 3 shows an example of an activity with 3 group sandboxes.

To assign a student to a certain group the teacher may drag participants and drop them over group sandboxes. In order to allow fast configuration of groups, especially in those cases where the groups have to be conformed with students having different levels of skills the teacher may use the "Randomize groups" functionality, by which students are randomly distributed to the groups trying to create groups of the same size. Group sandboxes display "participant icons" in order to represent the number of members. When students are assigned to a group, they are notified and a message displays the id of the assigned group. Other group members will be displayed on the student's interface by an icon and the name.

In order to manage students from a group, the teacher must "enter" into a group's sandbox by double clicking it. This will zoom in and display the detailed information of the visualized group, including members' icons with their name labels. Reconfiguration of the groups is always possible by dragging the icon of a student out from the original group sandbox to another.

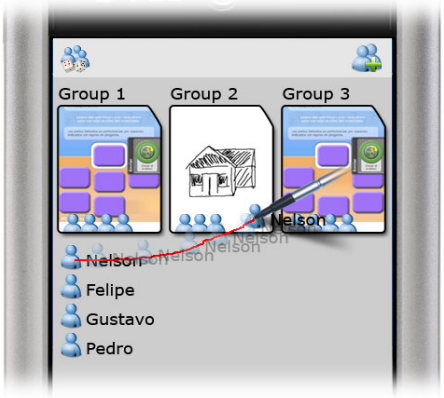

Figure 3: Teacher's module, in Group setup mode

Problems management: In order to allow teacher to create and distribute the problems students have to work on the teacher's module implements a "Problem manager" mode. This mode allows teachers to both create problems, using pen-based sketching and gestures, and load previously created problems. To create a new problem, the teacher must click the "New

problem" icon. The system will create the new problem icon, and zoom in, entering the "Problem edit" mode. Teachers can also load previously created problems by clicking the "Previous problems" icon.

Problem edit mode: In this mode, teacher can define the nature of the problem. There are two main types of problems: open and closed. Closed problems are more suitable to be used with a larger number of students since the feedback can be made automatically without the direct intervention of the teacher during the class. On the contrary, open problems allow the teacher to specify a richer variety of them. It is also easier and faster to change the specification of an open problem during the class. However, open problems require the direct intervention of the teacher for feedback and assessment.

Open problems are based on written or drawn instructions, which students will follow once problems are assigned. Closed problems consist on choosing the right option from a list of possible answers. For open problems, teachers are not required to configure possible answers sets and correctness assessment must be performed manually. For open problems students write or draw a solution.

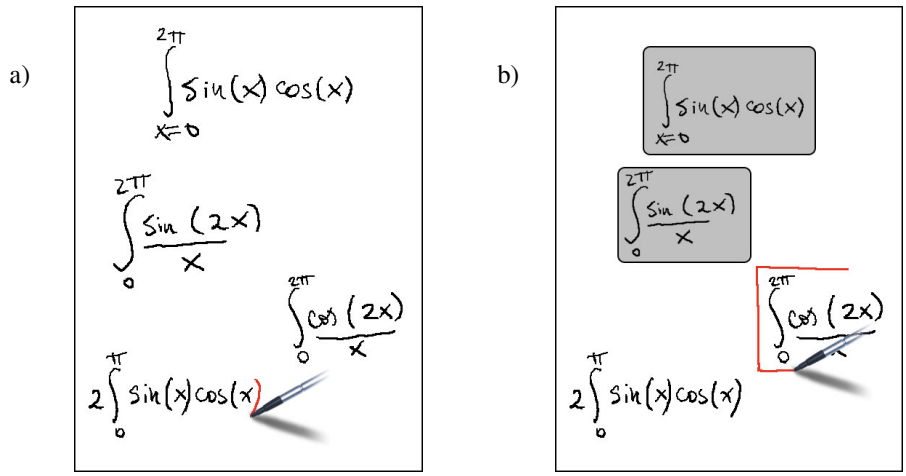

Figure 4: a) writing the problem definition and the answers; b) delimiting the elements by closing them in rectangles

For closed problems students must chose the right alternatives. There are three subtypes of closed problems: Single answer, multiple answers and sequence answer, (see "Answers assessment" section). Closed problems include a drawn or written description of the problem and a set of alternatives. The teacher defines the correct answer in order to enable automatic assessment (see "Problem answer" section). In order to create a problem's description and alternatives, the teacher must type, hand-write or draw texts and images. A closed problem consists of three parts: the problem description, the right answer and a set of wrong answers. Figure 4 shows this process. On the figure 4a) the teacher's hand-writing of the different elements of the problem is shown. The figure 4 b) shows how the elements can be defined and delimited by closing them into rectangles. The rectangle gesture is interpreted by the

system as a special instruction, which defines an element containing the strokes surrounded by the rectangle define by the gesture. Once the parts of the problem have been defined, the teacher must drag problem description into the description area, correct answers must be dragged to the solution area (more than one right alternative is allowed) and wrong alternatives must be dragged to the alternatives area. This is shown in the figure 4 c).

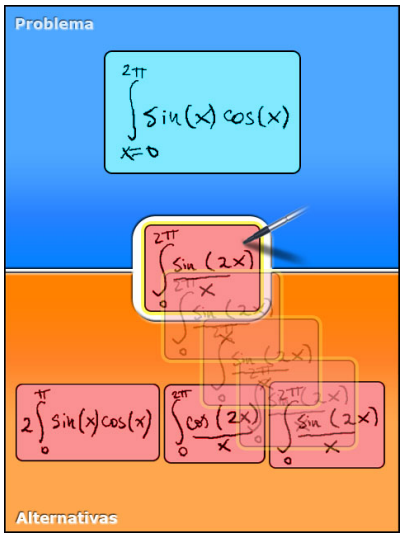

Figure 4: c) dragging the problem parts into the respective areas

Problem distribution: Once created, problems' have to be distributed to the various groups. For each problem an icon is displayed in the "problem manager" mode. The teacher can assign a problem to groups by dragging its icon into the group's sandbox. One problem may be assigned to several groups and one group may receive several problems. After the problem is assigned, the teacher may enter the "problem edit" mode to modify it or to create a new version of the same problem. This means, new problem proposals can be created starting from the existing ones, or that the same problem can "evolve" from simple to more complex challenges.

Problem answer: In order to show the students the problems that have been assigned to their groups the icons are shown in their devices as a list. A small figure shows if the assignment is pending, it is being worked, it has been completed and evaluated, or is waiting for the teacher assessment (for open problems). In order to answer a problem, students have to double click its icon, accessing the assignment in full screen. For closed problems the problem description is located in the upper region and the alternatives are located in the lower region. The middle zone of the screen varies depending on the subtype, displaying instructions for students. These instructions describe if students must choose a single answer, define a group of alternatives or define a sequence among the alternatives. First, each single member of a group must answer the problem individually, dragging the alternatives from the bottom area into the middle region.

Closed answers: In order to promote discussion and interaction among the member of a group, closed problems require all members of a group to agree on the correct answer before submitting it to the teacher. Once each student has defined her/his choice she/he must click the "Share answer with group" button, located in the "interaction palette" (Figure 5). As other members chose their answers, coordination lights turn red or green. For each member of the group, a light will turn green when the other student's answers coincide or red if not. Members who have not chosen their answers yet have all their lights off (grey). This tool allows students to coordinate their answers, to encourage debate, and to empower majorities.

Figure 5: The figure in the right shows the students must chose an answer by clicking on the "Share answer with group" button. After choosing an answer, the button turns into and agreement indicator (center). Here, the button shows the student's answer coincides with one student but differs from two other. Once all members agree on the answer, it turns into the "Submit answer" button (right),

Collaborative answer to open problems: Open problems allow students to write and draw their solution (Figure 6). Collaboration allows students to achieve a common answer, based on all group members' participation. For these problems students share a common writing/drawing area, where they can build a collaborative solution based on pen-based gestures.

Submitting Answers: Once the group has agreed about an answer (using the coordination indicator for closed problems or drawing the answer together in open problems), all members must agree to submit the problem to the teacher. Using the same interaction palette shown in the figure 4, each group participant must click the "Submit answer" button. Again, coordination lights are displayed to represent how many members have agreed to submit the solution. Once all members have agreed about submitting an answer, it is sent to the teacher.

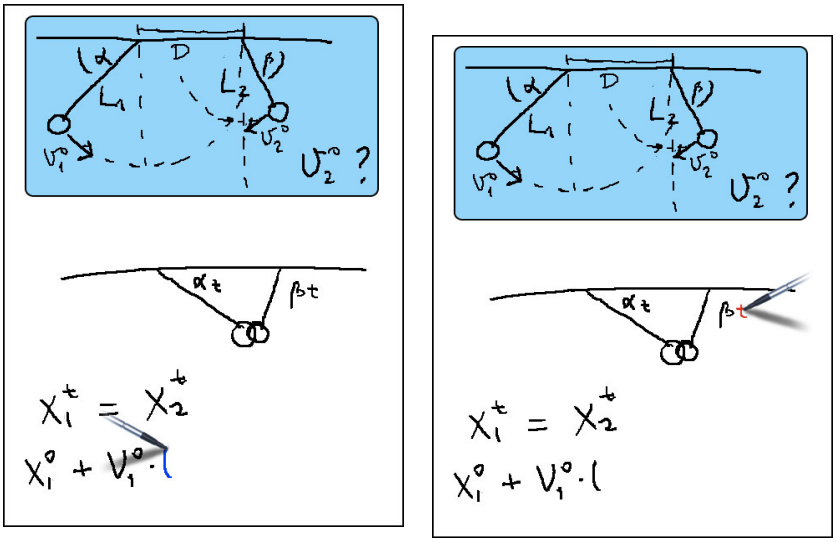

Figure 6: Screenshots of two students' PDAs jointly working in the same group work solving different part of a problem

Close problems' instant assessment: Because closed problems have defined answers, the system can automatically assess a group's answer to a certain problem. Therefore, as soon as the group agrees to send a closed problem answer, the system checks its correctness and sends a notification to the group. Students can immediately see the evaluation in the problem list.

Answer update: As far as an answer has not been submitted students can change their choice for the correct answer. For of closed problems, this would require the repetition of the solution agreement and the submitting process.

Online-assessments: As described in the section "Group Setup" of this chapter, the teacher can visualize the current activity of all groups at the same time (figure 3). This helps the teacher to check whether groups are working correctly, or to find out if any group may need feedback or further assistance. Once the teacher realizes a group needs help, she/he may enter the group's sandbox and input the feed back directly into the current working area of all participants. In this way, Students receive teacher's feedback in real time.

Activity results: The Activity Results mode of the teacher's module shows all assigned problems as a table containing reduced views of the answers where the teacher can check all answers at the same time. The teacher can check the solutions to any problem by clicking on any problem preview. Figure 7 shows an example of the activity results table. Tick and cross icons allow the teacher to quickly view the current state of the activity. Clicking again on any problem preview allows the teacher to provide feedback to students.

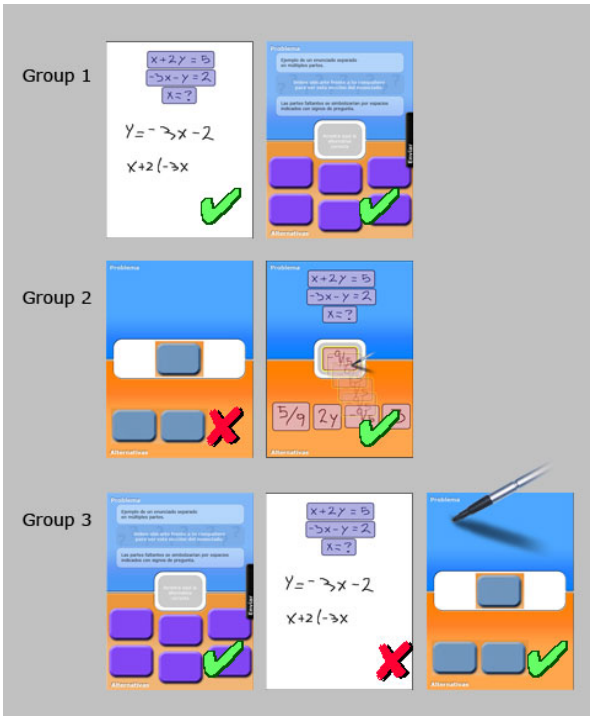

Figure 7: The teacher' view showing the results mode

5 The MCI Supporter Architecture

MCI Supporter has been developed on top of a framework previously developed by the authors of this work for supporting the development of other mobile applications for different purposes [Baloian, 2007] Some of them have been collaborative face-to-face design [Zurita, 2008] and meetings [Zurita, 2006]. The framework supports the development of applications in two ways. The first one refers to the programming of the communication functionalities of peer-to-peer applications and offers support at three levels: peers discovering, information exchange and group management. The bottom level is the discovering of peers, which is implemented using multicasting: each application sends multicast packages to announce its participation. All other applications will notice the presence and establish a TCP/IP communication channel in order to share information. This process is transparent for the application programmer and is initiated just by the fact of using the platform. The second level is the information exchange, which is implemented as a library of classes implementing what we called Shared Objects. A Shared Object is an abstract class which should be extended in order to create an object class whose state will be transmitted to all active participants when the object changes its state, this is, when one or

more object variables change their value. The programmer implements a shared object by extending the SharedObject abstract class. The third level is tightly coupled with the level implementing the Shared Objects and is the Application Group Management level. It was mainly the in-classroom learning scenario that motivated the development of this functionality as part of the framework because of the need to define subgroups of partners inside the whole group of active participants. For example, the teacher may want to propose a task which should be accomplished by small groups which do interact among them, but she wants to keep the possibility of looking what the different groups are doing. For this we developed the necessary mechanisms inside the middleware in order to create groups and have applications join and leave them. Any shared object created inside an application within a group will be also created in all the applications belonging to the group. If the state of the of the shared object changes, this will be transmitted to all applications of the group.

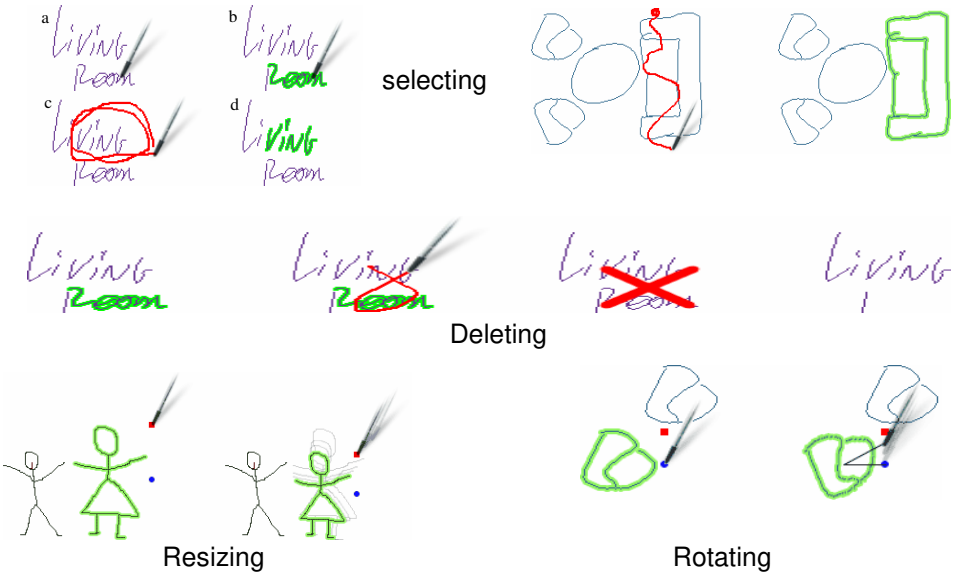

Figure 8: Some of the most interesting gestures supported by the platform in order to develop gesture-based application interfaces

The second the framework supports the development of applications for mobile devices is by providing the programmer with a class library that supports the development of application interfaces where user-machine interaction is based on gestures. This is especially convenient for implementing interfaces in small touch sensitive screens. Interaction based exclusively on gestures, minimizes the number of widgets and the need of a virtual keyboard and maximizes the space available for entering content when it consists of exclusively free handwriting inputs. Some of the most important gestures supported by the framework are described in figure 8.

6 Preliminary testing

The main hypothesis of this work is that MCI-Supporter mobile nature does take advantage of the mobile situation to promote social interaction inside the classroom environment. To test this hypothesis a large scale experiment is needed. However, the system has already been preliminary tested in a real scenario in order to test the suitability of the system to be used in the classroom and have some preliminary results about its usability in order to prepare it for a larger scale formal testing. During two weeks, 24 students and two different teachers from a pre-graduate university course used MCI-Supporter. The activity was aimed at exercising concepts learnt during course about software development. All type of problems where used during the experience as well as the group reconfiguration functionality. They used the system twice a week in sessions of one and a half hour each. As far as possible, all learning activities described in table 1 were tried. Teacher used a Tabled PC and the students used PDAs.

Figure 9: On the right pictures we see students belonging to one group working together during the preliminary test On the left pictures students during reconfiguration of the groups

During the activities we could observe high levels of social interaction for all the pedagogical activities. The teacher could easily and swiftly perform the activities of forming groups, distribute exercises and assess the student's work while maintaining the face-to-face communication with them. We also noted that group reconfiguration and inter-group interaction was eased by the fact that they were using mobile devices. We conducted a survey among the students in order to know their opinions about the usability and effectiveness of the system for supporting their work. 67% "agreed" that the system was easy to learn and to use, 33% "strongly agreed". 75% of the students "agreed" that they felt comfortable using sketches, gesture based interaction and overall with the visual metaphor the interface was based on. Another 25% "strongly agreed" with the same sentence. Regarding the effective support the system offered to boost the social interactions 66.6% "agreed", a 16.6% "strongly agreed", and a 16.6% took a neutral position. All students agreed that they felt more motivated to participate in learning activities supported by mobile computing since mobility enabled the face-to-face interactions with other students and with the teacher. Regarding the teachers, both agreed that the system can be applied to any type of content and that its operation is on the whole simple. Some comments from the teachers and

the students revealed that the collaborative editing of the sketches was difficult at the beginning and that they had to "learn" how to synchronize themselves. For this, face-to-face interaction was very helpful. They recommended use to improve this part of the system by incorporating more awareness hints.

7 Conclusions

The goal of MCI-Supporter is to support a pedagogic style that turns traditional face-to-face teaching into more of a two-way conversation between instructor and student. One of the most important design principles was to combine computer-mediated with face-to-face interaction for a more motivating learning environment. With this design principle in mind, the mobility of students and teacher turned into a requirement, and the problem-based and challenge based learning style was used in order to actively engaging students in the learning process. Although there are other works using also mobile technology to engage students in problem-base learning activities, we can conclude from the literature that this system is unique in the following aspects:

- It is a full peer-to-peer architecture, not using a central server, which allows the system to be used without restriction in any scenario, for example, outside the classroom in a laboratory, a museum, an exhibition to support learning "in the wild".
- Since creation of problems (and its solutions) is based on free-hand writing and sketching input, the system is independent of the subject being taught (as far as the teacher can imagine problems based on sketches and text) and it allows the creation of new problems with open or closed answers "on the fly". During the preliminary test, we saw this feature to be used by one teacher to modify the original problem several times, thus the problem evolved stage by stage to a more complex one.
- It allows the teacher to overview all the workspaces of the groups at the same time. The teacher can choose to give immediate feedback (or proposing a new problem) to any of the groups by joining the synchronized workspace of that group.

Finally, we would like to stress that, although mobile technology represents a big potential for supporting students' learning, they have important limitations. Therefore mobile PDAs-based applications must be carefully designed to account for these limitations.

Acknowledgements

The work of this paper was funded by Fondecyt project 1085010 and a NICT scholarship.

References

[Anton, 1999] M. Anton, The Discourse of a Learner-Centered Classroom: Sociocultural Perspectives on Teacher-Learner Interaction in the Second-Language Classroom, The Modern Language Journal, 1999, 83(3), 303-318.

[Baloian. 2006] N. Baloian, K. Hoeksema, U. Hoppe, M. Milrad: Technologies and Educational Activities for Supporting and Implementing Challenge-Based Learning. Education for the 21st Century: Impact of ICT and Digital Resources. Deepak Kumar & Joe Turner (Eds.) Springer, NY, USA, 2006, 7-16.

[Baloian, 2007] N. Baloian, G. Zurita, P. Antunez, F. Baytelman: A Flexible, Lightweight Middleware Supporting the Development of Distributed Applications across Platforms. Proceedings of the 11th CSCWD conference, Melbourne, Australia, IEEE press, pp. 92-97, 2007

[Baloian, 2008] N. Baloian, J.A. Pino, H. U. Hoppe: Dealing with the students' attention problem in computer supported face-to-face lecturing. Educational Technology and Society 11(2), 2008, 192-205.

[Barab, 2000] S. A Barab & T. Duffy: From practice fields to communities of practice. In: D. H. Jonassen, & S. M. Land, (Eds.), Theoretical foundations of learning environments.). Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 2000, pp. 25-56.

[Barkley, 2005] E.F. Barkley, K.P Cross, C.H. Major, Collaborative Learning Techniques, Joseey-Bass, San Francisco, 2005.

[Belamy, 1996] R.K.E. Bellamy, Designing Educational Technology: Computer-mediated Change. In Nardi, B. A. (Ed.). Context and Consciousness: Activity Theory and Human-Computer-Interaction (pp. 123-146). Cambridge, 1996, Massachusetts: MIT Press.

[Bravo, 2005] J. Bravo, R. Hervás, G. Chavira, Ubiquitous Computing in the Classroom: An approach thorough Identification Process, Journal of Universal Computer Science, 11, 9, 2005, pp. 1494-1504.

[Brown, 1989] J. Brown, A. Collins & P. Duguid: Situated cognition and the culture of learning. Educational Researcher, 18, 1, 1989, pp. 32-42.

[Collazos, 2003] C. A. Collazos, L. Guerrero, J. A. Pino, S. F. Ochoa: Improving the Use of Strategies in Computer-Supported Collaborative Processes. Lecture Notes in Computer Science 2806, 2003, 247-260.

[Cotton, 1998] K. Cotton,: Monitoring student learning in the classroom. School Improvement Research Series Close-Up #4, 1988.

[Grüntgens, 2004] W. Grüntgens and R. Melzer, Diagnostische kompetenz von lehrkräften im primar- und sekundarbereich. http://pz.bildung-rp.de/pn/pb1 04/diagnostischekompetenz.htm, Last visited: August 2008.

[Johnson, 1996] D. W. Johnson & R. T. Johnson: Cooperation and the use of technology. In D. H. Jonassen (Ed.), Handbook of research for educational

communications and technology, pp. 1017-1044. New York, 1996, Simon & Schuster Macmillan.

[Koshmann, 2005] T. Koschmann, A. Zemel, M. Conlee-Stevens, N. Young, J. Robbs and A. Barnhart, How do people learn? Members' methods and communicative mediation, In R. Bromme, F.W. Hesse & H. Spada, 2005.

[Liu, 2003] T. Liu, H. Wang, T. Liang, T. Chan, W. Ko and J. Yang, J. "Wireless and mobile technologies to enhance teaching and learning", Journal of Computer Assisted Learning, 2003, 19(3), 2003, 371-382.

[Liu, 2002] T. Liu, H. Wang, T. Liang, T. Chan, and J. Yang, "Applying wireless technologies to build a highly interactive learning environment", In proceedings of IEEE International Workshop on Wireless and Mobile Technologies in Education, WMTE, Sweden, 2002.

[Marcelino, 2007] M. J. Marcelino HME: a Handheld Model Editor for Educational Contexts, Journal of Universal Computer Science, 13, 2007, pp. 1012-1021

[Müldner, 1996] T. Müldner, R. A. Nicholl, Computer-Supported Human Cooperation in Electronic Classrooms, Journal of Universal Computer Science 2, 10 1996, pp. 679-693.

[Molina, 2005] A. Molina, M. Redondo, M. Ortega, A System to Support Asynchronous Collaborative Learning Tasks using PDA's, Journal of Universal Computer Sciences, 11, 9, 2005, pp. 154-1554.

[Resnick, 1987] L. Resnick: Learning in school and out. Educational Researcher, vol 16, (9), 1987, pp. 13-20.

[Roschelle 2003] Roschelle, J.: "Unlocking the learning value of wireless mobile devices", Journal of Computer Assisted Learning, 19, 3, 2003, pp. 260-272.

[Roberts, 2006] J. Roberts Pervasive Health Management and Health Management Utilizing Pervasive Technologies: Synergy and Issues, Journal of Universal Computer Science, 12, 1 (2006), 6-14

[SCRG, 2005] Social Computing Research Group, University of Missouri-Columbia: Understanding computer mediated social experience: implications for CSCL, Proceedings of th 2005 conference on Computer support for collaborative learning, Taipei, Taiwan, pp. 617-621

[Skinner, 1993] E.A. Skinner and M.J. Belmont, Motivation in the Classroom: Reciprocal Effects of Teacher Behavior and Student Engagement across the School Year, Journal of Educational Psychology, 1993, 85(4), 571-581.

[Sturm, 2005] J. Sturm, R. Iqbal, O. Kulyk, C. Wang, J. Terken: "Peripheral Feedback on Participation Level to Support Meetings and Lectures", in Proceedings of Designing Pleasurable Products Interfaces (DPPI), Eindhoven, The Netherlands, October 2005, pp. 451-466.

[Suthers, 2005] D. Suthers, Technology affordances for intersubjective Learning: thematic Agenda for CSCL, T. Koschmann, D. Suthers and T.W. Chan. (Eds.) Computer Supported Collaborative Learning: The Next 10 Years!. Mahwah, NJ:

Lawrence Erlbaum Associates, International Society of the Learning Sciences, 2005, 662-671.

[Vygotsky, 1978] L.S. Vygotsky, Mind in society: The development of higher psychological process, Cambridge, MA: Harvard University Press, 1978.

[Wenger, 1998] E. Wenger: Communities of practice: Learning, meaning and identity. Cambridge, UK, 1998: Cambridge University Press.

[Wright, 1995] A.W. Wright: Teaching Improvement Practices: Successful Strategies for Higher Education, Anker Publishing Co, MA, 1995.

[Zurita, 2007] G. Zurita, P. Antunes, N. Baloian, F. Baytelman: Mobile Sensemaking: Exploring Proximity and Mobile Applications in the Classroom, Journal of Universal Computer Science 13, 10, 2007, pp. 1434 - 1448

[Zurita, 2008] G. Zurita, N. Baloian, F. Baytelman: 2008, A Collaborative face-to-Face Design SupportSystem based on Sketching and Gesturing. Advanced Engineering Informatics. Vol 22, Issue 3. July 2008, pp. 340-349.