Lightweight Extensions of Collaborative Modeling Systems for Synchronous Use on PDA's

Niels Pinkwart, Christian Schäfer, H.Ulrich Hoppe
University of Duisburg
{pinkwart, schaefer, hoppe}@collide.info

Abstract

Based on experience in orchestrating collaborative learning scenarios with ubiquitous computing technology, two strategies for extending a co-constructive modeling environment with PDAs connected through a wireless LAN are described. One application is an annotation tool, the other replicates the modeling system on the PDA, both provide full synchronization with the PC environment. General implementation strategies for such extensions are discussed.

1. Introduction: Ubiquitous computing in integrated learning environments

Our interest in introducing PDAs in educational scenarios originates from the idea of using ubiquitous computing elements (cf. [1], [2]) to orchestrate classrooms with embedded interactive technologies in an unobtrusive way. Our idea of a "computer-integrated classroom" has been practically elaborated and put into practice in the European NIMIS project (1998-2000, cf. [3]). The most evident and concrete result of NIMIS is a classroom installation at a primary school which features special hardware such as an interactive whiteboard and pen-based tablets embedded in the pupils desks in a networked environment with educationally motivated groupware functions. The software includes a special application for initial reading and writing ("Today's Talking Type-writer") using pen-based input and speech synthesis, as well as a special desktop which facilitates archiving and communication functions for early learners even before they are skilled in reading and writing. The NIMIS classroom is still in everyday use. Scenarios using similar ubiquitous computing elements have also been installed for practical use in our academic teaching [4].

In the development of these scenarios, we have formulated and applied the following principles:

- provide uniform access to multiple representations of media and use a variety of information sources;
- do not let the technology "get in the way" but facilitate existing classroom procedures;
- do not let the educational scenario be determined by the use of a computer, but let interactive digital media be a "resource at hand" in the background similar to the traditional use of paper and pencil;
- exploit the value added from being able to easily replicate, distribute, and re-use externalized learning results in a networked digital environment.

Figure 1: The NIMIS classroom

More recently, we have focused our attention on combining technologies for synchronous collaboration in shared workspaces with interactive-constructive environments based on computational representations resulting in what we call "collaborative mind tools" [5]. Typical examples are collaborative modeling environments, e.g. based on "System Dynamics" or "Petri Nets".

As a consequence of the NIMIS and other experience, we are convinced that pen-based interaction with multi-representational software is a key factor for making digital or computational technologies flexibly available without dominating the educational scenario. Yet, the NIMIS scenario uses wired devices with fixed locations in the physical environment. Using wireless networking would extend the range of classroom scenarios and processes to be served, and it would allow for making results directly "physically portable", also between different locations (e.g. school, the "field", and home). PDAs appear to be a straightforward solution. In the sequel we will describe and discuss our approach and experience to incorporate PDAs.

2. Educational applications of PDAs

The currently available educational applications on PDAs can be categorized according to two main types of usage:

a) the PDA serving as an interface to a "main" desktop program to extend the use of the desktop application for specific scenarios; here, the mobile device may in the extreme case just serve as a front end, e.g. for outdoor data input;
b) a standalone application running on the PDA, with or without connection to a central desktop application; this approach includes also several mobile applications allowing collaboration via direct communication between the devices.

For the first category, "ImageMap" from SRI International, or the "museum guide" of CILT [6]. In the case of "ImageMap", the PDA is used by students who receive an image on the mobile device and have to answer a given question concerning the image. Having done so, they send their annotations back to a server where all the different comments are gathered and displayed on a public screen allowing the teacher as well as students to discuss the answers.

Similar to the case of "ImageMap", the mobile application "museum guide" is also essentially an interface for communication with a central server. It is used primarily for retrieving data and displaying information about a museum. Also, the current location of a user can be detected and is considered for offering location-based information to the user, e.g. a museum or trade fair visitor.

Applications and concepts illustrating the second category include "Geney" by EDGE Lab and CS Division and "PiCoMap" from the hi-ce group [7]. The goal of "Geney" is to collaboratively "engineer" a fish with a particular set of characteristics under restrictions coming from genetic rules. The students take different roles: one of them acting as a "manager" whose fish will be paired with one fish collaboratively constructed by the other students. During a so-called "what-if" mode, the view on the mobile applications differs according to the student's role: the manager sees a condensed overview and whereas the other participants have a more detailed but restricted view of resulting characteristics. So, the students have to combine perspectives and collaborate to achieve optimal results.

With the "PiCoMap" application, students illustrate a specific problem using a graphical representation consisting of nodes, text input, and directed links. Having done so, they exchange their models pair-wise using infrared, and afterwards they annotate the ideas of the co-learner. The aim of this system is to lead students to a discussion about their various ideas and different views and, finally, to a revision of their original ideas.

Most of the mentioned tools use infrared connection as the channel to exchange information between mobile devices. The disadvantages of this approach are that it does not directly support continuous co-construction in shared workspaces (instead, only repeated "one-time" data upload or download is facilitated) and that it is quite restricted in terms of bandwidth. While the second aspect is of limited relevance in usually not over-complex pedagogical scenarios, the first disadvantage restricts the spectrum of potential collaborative processes. Using wireless LAN connections can solve this problem and thus offer more flexible ways to support collaborative work. Thus, completely synchronized mobile applications are enabled for a variety of collaborative scenarios.

In the next section, we will present different approaches to extending existing co-operative modeling environments with mobile devices, especially with PDAs. Here, completely replicated and fully synchronized applications do not make much sense due to the limitations of the PDA or of the available bandwidth. Instead, we favor *lightweight* integration strategies which can be formulated following the model-view-controller concept:

- a partial view on the general application state, especially considering the screen size;
- a reduced processing functionality (control), adapted to the device and its I/O capabilities;
- partial data (model), taking into account memory and processing restrictions of the mobile device.

The "partial view" principle originates from the physical-geometrical restrictions of PDAs and could potentially be relaxed or overcome by using bigger handheld devices such as e-book readers. Currently, a disadvantage of these devices is the higher cost and unclear future in the market. A general disadvantage shared by PDAs and e-book readers is that they usually do not run the same system platforms as PCs or workstations. This requires costly re-programming. The three mentioned "lightweight" principles may also guide this kind of re-programming.

3. A platform for collaborative modeling

The synchronization mechanism that we use in a variety of collaborative systems is Java MatchMaker TNG [8]. It is built upon Java RMI and basically consists of a central server with clients which can also run on the currently available "small" Java environments for mobile devices (as, e.g., PersonalJava for Windows CE). Different from e.g. NetMeeting [9] or other centralized approaches, MatchMaker works with replicated applications whose objects can be partially or fully synchronized in a flexible way. The typical applications are shared workspace environments. As a consequence of the replicated architecture, each application instance can hold a mix of private and shared workspaces between which structured data can be easily transferred.

MatchMaker TNG arranges the synchronized data in a tree, allowing clients to listen to changes on arbitrary sub-trees. This facilitates, e.g., the partial coupling of applications. This option is of special relevance for PDA's or other non-standard computing devices as it can be used to achieve the desired lightweight integration without a loss of information for the other applications.

Cool Modes (*CO*llaborative *O*pen *L*earning and *MODE*lling *S*ystem) [5] is a collaborative tool framework designed to support structured discussions and co-operative modeling processes in various domains. Like in some other environments such as Belvedere [10], this is achieved through a shared workspace environment with synchronized visual representations. A special feature of Cool Modes is that it does not use a predefined built-in representation, but different "visual languages" can be easily specified and made accessible on the collaborative Cool Modes platform as plug-ins (or "palettes"). A palette specifies the basic lexical and syntactic elements of the language in terms node and link types. Operational semantics can be added through specific interpreters. Such interpreters are currently available for Petri nets and different mathematical models (including stochastics and system dynamics). A Belvedere-like language for argumentation graphs is only defined on the syntactic level. The different languages can be mixed in the same workspace and they can additionally be annotated using pen-based input. The flexibility of mixing different visual languages and annotations allow for using Cool Modes as a tool "at hand", in the same way as paper and pencil.

As a standard feature, Cool Modes allows the use of multiple workspaces represented in different windows which can be arranged freely. Each workspace consists of a number of transparent layers which can contain "solid" objects such as, e.g., handwritten strokes, images and other media types. Four predefined layers with different functionality exist by default – one for a background image, one for annotations, and two for other objects.

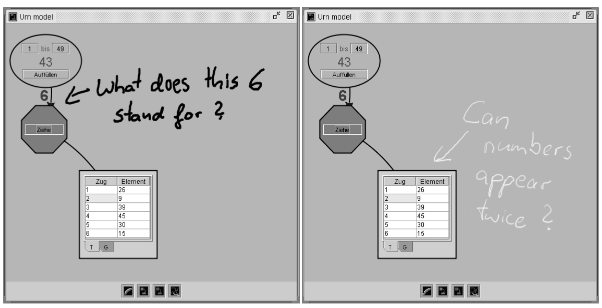

Figure 2: Partially synchronized workspaces

The built-in cooperation support in Cool Modes basically relies on the provision of synchronously shareable representations based on the MatchMaker TNG server. The underlying synchronization tree reflects the logical structure of the application (workspace, layer, and objects). Accordingly, flexible partial coupling is possible by workspace or by layer (as shown in figure 2 with an example from "stochastic experiments") or even between single objects. Depending on the concrete scenario, each of these options can be useful, e.g. private handwriting layers in synchronized workspaces or the sharing of model parts without "publishing" the whole model.

4. Extensions for hand-held devices

The principal approach for integrating mobile devices into the Cool Modes framework was based on the premise to generally allow for synchronous hand-written input from the PDAs on a dynamically added user-specific annotation layer and to have the rest of the lightweight synchronization dependent on the concrete scenario. In the one case presented, only an image of the Cool Modes workspace is transmitted to the PDA; in the other case, a specially adapted "small" modeling tool runs on the mobile device, completely synchronized with the other environments (desktop or mobile).

The first scenario and tool was motivated by our practical experience with presentation and group scenarios in academic teaching [4]. Several of our lecture halls are equipped with an electronic whiteboard that can be used by the teacher instead of the traditional chalkboard. Thus it allows for the free-hand exposition of ideas, both written and in the form of sketches, but it also allows for using computerized modeling tools, which is an ideal combination in many areas of science and economy.

The (digital) interaction with the students is a crucial point here – we believe that the possibility of making private annotations to the public slides developed by the teacher *at the time these are created* would often be beneficial. The teacher might also want to allow the presentation of some student's annotations on the whiteboard or to initiate certain collaborative tasks among the student group following his presentations.

This is realized by "CoolCom", a local annotation tool implemented under Windows CE with PersonalJava v1.1. The CoolCom window is coupled with a subregion of a Cool Modes workspace and shows the current content as a background image which is annotated in different colors. CoolCom has the following characteristics:

- the local view consists of 320x240 pixels (scrollable) so that only parts of the workspace can be annotated;
- in contrast to the main application, CoolCom just allows free-hand input (as strokes with color);
- in CoolCom, the Cool Modes graph of linked objects (JGraph) is reduced to a background image.

In detail, the synchronization between Cool Modes and CoolCom is realized as follows: the CoolCom application on a Compaq iPAQ joins a MatchMaker TNG session and selects a specific Cool Modes workspace to be annotated. In Cool Modes, when this is detected, a specific layer for the hand -written annotations is dynamically created and completely synchronized with the corresponding CoolCom instance. All the other layers of the workspace in the "main" application are *not* directly coupled (these are usually too large concerning file size and therefore transmission time in the WLAN). Instead, they are captured as images in regular intervals. These images are sent to CoolCom and displayed as background images so that they constantly mirror the main application. They can be annotated by users on the mobile devices, even by several users at a time. Due to the layer structure, these local annotations done on the PDA can be turned visible or invisible in Cool Modes according to the requirements.

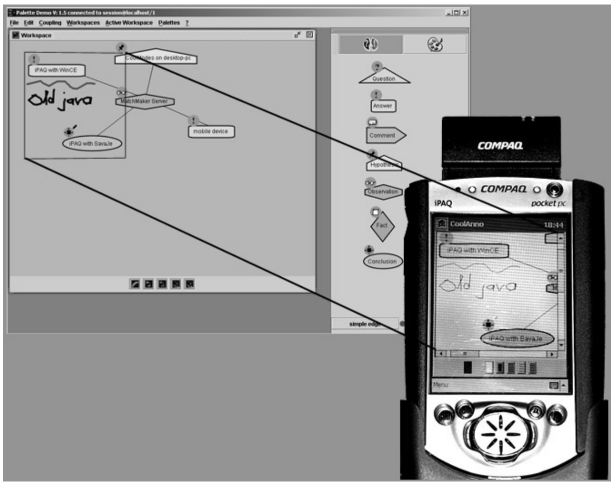

Figure 3: Annotating Cool Modes with CoolCom

With the second scenario, we support regular school lessons in mathematics, especially an introduction to probabilities using the "stochastics" extension of Cool Modes (cf. figure 2.) An integral part of the activities in this area are simulated and real stochastics experiments and the comparison and aggregation of the results from different learning groups. Here, PDAs allow us to provide computational support in the classroom with minimal changes in the physical scenario. We only assume that there is one workstation or PC connected to a big interactive screen. PDAs can also be used for annotations, and they can be easily used together with dices or other physical objects.

This mobile application, "microUrn", is currently being developed on SavaJe, a pure Java OS for the iPAQ that supports Swing and most other classes of JDK 1.3. Here the PDA holds a complete data model, yet several data collections may be merged from different groups. All data are stored in a standard XML format generated by Cool Modes. Results are shared with the big screen or between groups using the synchronization mechanism. The local application on the PDA works with a partial view and limited functionality.

5. Conclusions

We have exemplified two different strategies for supporting synchronous cooperation with visual modeling environments, one without "deep" sharing of data and the other with a shared data model. Both applications are restricted in terms of view and processing aspects. Processing limitations may be overcome, but not every possible function makes sense on a small device which is frequently switched on and off (e.g., group archival functions). We appreciate that PDAs are really "at hand" without binding the attention of the learner too much. Yet, the view limitation is inherent in the definition of PDA as a compact "pocket device". We consider it worthwhile to look for alternative wireless and pen -based devices with bigger displays and similar portability.

6. Acknowledgements

Parts of this work have been supported by the European IST project No. 2001-25214, SEED.

7. References

[1] M. Weiser (1993). "Some Computer Science Issues in Ubiquitous Computing", *CACM* 36 (7), 75-84.
[2] D.A. Norman (1998). "The invisible computer", Cambridge (USA): MIT Press.
[3] Hoppe, H.U.; Lingnau, A.; Machado, I.; Paiva, A.; Prada, R.; Tewissen, F.: Supporting collaborative activities in computer-integrated classrooms – the NIMIS approach. In Proc. of *CRIWG 2000*. Madeira, October 2000 (IEEE Press).
[4] Hoppe, H.U.; Luther, W.; Mühlenbrock, M.; Otten, W.; Tewissen, F. (1999). Interactive presentation support for an electronic lecture hall – a practice report. In Proc. of *ICCE '99*. Chiba (Japan), November 1999.
[5] Pinkwart, N., Hoppe, H.U., Bollen, L. & Fuhlrott, E. (2002). Group-oriented modeling tools with heterogeneous semantics. To appear in *Proceedings of ITS 2002* (Biarritz, France, June 2002). Berlin: Springer.
[6] Roschelle, J., Pea, R. (2002). A walk on the WILD side: How wireless handhelds may change CSCL. In *Proceedings of CSCL 2002* (pp. 51-60). Boulder (USA), January 2002.
[7] Luchini, K., Quintana, Ch., Curtis, M., Murphy, R., Krajcik, J., Soloway, E., Suthers, D., (2002). Using handhelds to support collaborative learning. In *Proceedings of CSCL 2002* (pp. 704 - 5). Boulder (USA), January 2002.
[8] Jansen, M., Pinkwart, N. & Tewissen, F. (2001). Match - Maker - Flexible Synchronisation von Java -Anwendungen. In Klinkenberg, R., et al. (eds.): *LLWA 01 – Proceedings of GI - Workshop "Lernen -Lehren-Wissen-Adaptivität"*. Dortmund (Germany), October 2001.
[9] http://www.microsoft.com/windows/netmeeting
[10] Suthers, D., Weiner, A., Connelly, J. & Paolucci, M. (1995). Belvedere: Engaging students in critical discussion of science and public policy issues. In *Proceedinds of AIED 1995* (pp. 266-273). Washington DC (USA), August 1995.