Context awareness for application sharing in teaching environment

Cora Burger, Eric Bullinger, Stella Papakosta, Thomas Wagner

***Abstract*--Mobile and ubiquitous computing offer new possibilities to increase the awareness in class and support an active participation of students. To this end, context information is captured and distributed about sessions, participants and those applications that are used during a lecture or encounter among students (e. g. presentations, development and simulation environments, experiments in natural and engineering sciences). The framework of SASCIA (System Architecture Supporting Cooperative and Interactive Applications) cares for appropriate usage of this information to enable new ways of application sharing. First experiments have proofed the usefulness of the system and an acceptable performance.**

***Index Terms*—Application sharing, context awareness, teaching.**

I. Introduction

TEACHING achieves its best results, if students are inspired to actively deal with a topic. Hence, suitable facilities are needed and an appropriate atmosphere has to be created to let students experience the material by themselves, discuss and test different aspects, ask questions and make comments. In short, a constructive rather than an instructive manner of teaching is the preferred one. Moreover, learning in groups has to be encouraged to increase its effectiveness and to train social abilities.

Currently, such intensive teaching can be practiced in small classes only with all participants being close as well to one another as maybe to some experimental equipment. Difficulties appear as soon as one of these parameters does not hold any longer, that means in a large auditorium or with participants and experiment being separated geographically.

By transmitting audio and video from lectures to distributed groups of students, remote scenarios can be supported (e. g. [7] and [17]). This mechanism is worth while for people who would encounter problems and too much cost in traveling. But in general, tele-presence is not well accepted due to its lack of mediating the feeling of really being together. Moreover, tele-lectures have proofed rather poor with regard to interactivity and students becoming actively involved. Even new mechanisms and policies to provide more flexible access to microphone and keyboard, did not change this situation tremendously (cf. [19]).

In contrast to human interaction, there is a much higher motivation to perform laboratory experiments from remote. While it can provoke the most intensive experience for learners, to be close to an experiment and really watch the whole process, there are some serious reasons against it. Usually, equipment for experiments is quite expensive but nevertheless rather seldom in use. Hence, the possibility of sharing the same equipment in the form of tele-experiments among different, geographically separated institutions, increases the selection of available experiments for teachers and students. Furthermore, it is not always feasible and can even bear risks to be near, e.g. in chemical or nuclear plants. As a consequence, some monolithic solutions have been designed already to enable tele-experiments in a special teaching area for single users as well as for collaboration (e.g. [14] and [9]).

With more widespread usage of mobile devices and the upcoming technology of wireless networks, new possibilities are arising for collocated and remote ways of teaching. Currently, a lot of work is underway to bring wireless equipment to the campus and find out its potential for teaching. A first step is achieved by making available services like electronic mail, access to information and course material as well as notification throughout the whole campus [11]. More sophisticated features are offered by the following work:

- [1] provide a facility for remote pointing in application windows for a scenario among an expert and somebody working in the field. This possibility can be applied to teaching situations as well as shown by [23].
- Some systems ([16], [21]) examine further possibilities like the electronic pendant of hand raising and direct feedback to teachers via handheld computers during the lecture.

From a more general point of view, both cases have in common that some state information and its visibility to participants are involved. In literature, this fact is called

Manuscript received December 1, 2001. The work of the first author is funded by the Margerete von Wrangell scholarship of Baden-Württemberg, Germany.

C. Burger and S. Papakosta are with the Institute of Parallel and Distributed High-Performance Systems, University of Stuttgart Breitwiesenstr .20-22 D70565 Stuttgart, Germany (telephone: +49 711 7816-227/381, e-mail: {caburger, papakosta}@informatik.uni-stuttgart.de).

E. Bullinger is with the Institute for Systems Theory in Engineering, University of Stuttgart Pfaffenwaldring 9 D 70550 Stuttgart, Germany (telephone +49 711 685 7752, e-mail: eric@ist.uni-stuttgart.de).

T. Wagner is with the Institute of Industrial Automation and Software Engineering, University of Stuttgart, Pfaffenwaldring 47 D 70550 Stuttgart, Germany (telephone +49 711 685 7295, e-mail: wagner@ias.uni-stuttgart.de).

context awareness (e. g. [11], [17]): Either an individual is informed about its own state and its environment or other people are getting such information. This holds for the electronic as well as for the physical world. For instance, people can be informed about activities of other persons in an electronic shared workspace where members are allowed to bring in or modify folders and files for shared access [2]. Even in the physical case, more and more types of context information are becoming available like e.g. position, temperature, and intensity of light or noise. To this end, either cameras or suitable sensors can be used (e. g. [14], [24].

With respect to teaching, only examples as mentioned above have been presented up to now. But especially when thinking of tele-experiments in different teaching disciplines, a lot more information is of concern. In this paper, we are studying context awareness, its role for application sharing in teaching sessions as well as realization aspects in a systematic way. Based on two typical teaching scenarios, need and usefulness of context awareness are demonstrated in Section II. From these scenarios, a list of requirements is derived for application sharing in general including those for awareness in class. For the latter, a suitable model is built.

To really profit from context information, its handling has to be integrated into an overall framework to share teaching applications. To this end, we have designed the system SASCIA (System Architecture Supporting Cooperative and Interactive Applications, Section III). Besides components to capture context information, it is containing components for user and session management as well as generic frames to enable the plug-in of typical teaching applications. The latter are evaluating, distributing and taking into account context information in a suitable way (Section IV). In Section V we exemplify these concepts at three different cases from teaching in the areas of communication networks and engineering. The current state of the project and some first measurement results are described in Section VI. Outlining some directions for future work concludes the paper.

II. Requirements and Model

Two different scenarios, one with teacher and the other one without, are motivating our work.

A. Scenario A: Guided experiments during lectures

The first scenario concerns a scheduled lecture with teacher and students. We assume most of them to be collocated in the same room. All participants have to authenticate to proof that he or she is allowed to attend respectively act as a teacher. Typically, students are not always in time and can enter after the start of the lecture or leave in advance. Some cannot attend at all. Depending on the reason, students may want to participate remotely from the beginning in real time, maybe switch to the collocated case some time later on or at least replay the lecture afterwards at a suitable moment. Hence, the number and presence of participants can vary during such kinds of teaching sessions. Moreover, devices with totally different characteristics can be used for participation. Besides available resources and operating system, even the quality of the connection can differ [3].

Traditionally, teachers of all disciplines have been writing on a blackboard, showing physical slides or experiments or have been discussing a topic with their students. All these media have been used in a combined way. Nowadays, teachers are relying more and more on a designated computer that is equipped with a projector and suitable input/output facilities, e.g. a touch-sensitive screen. This opens up many possibilities like using a tool for presentation (e. g. the products PowerPoint, StarOffice or Acrobat Reader), demonstrate some engineering application or simulation (e. g. products like Jbuilder or Matlab), write and draw either on blank screen (ideal instrument to support discussions) or on projected application windows and so forth. Last but not least, even tele-experiments come into sight, where the teacher is controlling some equipment directly from the auditory as discussed already in the introduction. To this end, he must know exactly about the current state of all components involved into the experiment.

Thinking of students carrying mobile and handheld devices with them, opens up new possibilities for them as well, thus increasing interactivity during lectures. At first, students are enabled to download material as long as it is not due to copyright or license restrictions. Moreover, they should be allowed to not only perform private note taking but as well point and annotate on public windows as projected on the screen. For subsequent learning and preparation of exams, it is useful to record all these data by carefully obeying the correct temporal ordering and durations. Last but not least, even control of presentations and further kinds of applications by students should be possible. Among others, this holds for the loudspeaker of the room via the microphone of a student’s mobile device to make her understandable even in a very large auditory.

While originally the teacher will be having the right for such tasks, she should be able to grant it wholly or in parts to a student on request. Having controlled an application by more than one person at the same time increases the need for context awareness, i. e. complete information about the state of all components and actions of other participants.

To enable the teacher to get aware of questions, comments or general feedback of students and react accordingly, a mechanism is needed to identify such events. Depending on the size of room and crowd, there is a risk to overlook raised hands and be unconscious regarding the understanding of students. In remote settings, this problem is getting even more serious. Moreover, when addressing students one cannot exclude the possibility that they want to stay anonymous. Hence, they should be identified either by real name or by seat number.

Especially in a large auditory, learners may want to find their friends for further talks after the lecture. In some cases, it can even proof useful to select other participants and discuss topics in a manner that imitates whispering. This can lead to perturbation, therefore the teacher has to decide whether to allow or prohibit this functionality.

In general, teachers must be able to guarantee a proper performance of the lecture and control students' actions, especially revoke the right for application control from them at any time. In the worst case, even removal of students from the session should be possible. Furthermore, wireless connection to the rest of the world bears the risk of seducing students to do different things instead of following the lecture. Thus, some mechanism is required to prevent them from too much distraction.

B. *Scenario B: Learning and performing experiments without guide*

The second scenario takes place in a group of students without any supervising person being present. These students are discussing some topic (exercises, exams) and maybe are performing experiments. Their encounter can arise in a spontaneous or planned manner and need not take place in a room with a designated computer at hand. Instead, students can sit elsewhere inside or outside buildings without any such equipment. Like in the first scenario, they can participate with devices having different characteristics, in a local or remote manner with consequences regarding the connection quality. Hence, awareness can be achieved only when having at hand a list of all participants, their locations and resources.

After authentication, students should be able to access file servers on the campus and share applications including tele-experiments. Since all participants are on the same level, the problem can arise whom to assign the right to control the session or remove disturbing elements. Depending on the social abilities of participants, some kind of group decision support can be worthwhile. Moreover, a mechanism for surveillance and help can be required. Analogous to Scenario 1, further facilities like support for latecomers, public and private annotations to shared applications as well as recording and replay should be available.

C. *Requirements*

Both scenarios together are posing the following requirements (arranged in order of novelty):

- Awareness in class regarding participants, their identification, seat number, role, level of attentiveness and understanding, hand raising.
- Visibility of relevant application information to be able to control them in a proper way.
- Openness of the system, i.e. the possibility to plug-in applications from different teaching disciplines, going hand in hand with the requirement to integrate further services, especially those providing awareness information.
- During a session, a whole set of such applications should be usable enabling participants to switch among them.
- Different user groups and hierarchies (teacher, students) have to be coped with, resulting in differences in interaction styles: moderated session or scenario with peers, whispering allowed or disabled.
- Easy and quick set up of sessions to not bore teachers and learners but let them concentrate on the essential. Support for latecomers.
- Need for authentication and possibility to restrict membership.
- Role-dependent right to control participants.
- For authorized persons only: remote control of public loudspeaker, applications including their equipment, pointing and public annotation in application windows (as shown on each screen and by the projector).
- Access to teaching material including public annotation by taking into account copyright and license restrictions.
- Private annotation.
- Recording in such a way that real time replay of the whole presentation including audio, video, public and private annotation is possible afterwards.
- For usage at universities, the system must be independent from platform and should be available for free.
- Sufficient stability and security as well as prevention from distraction should be provided.
- User interfaces should be adapted to different natural languages with minimal effort. This is underlined by the fact, that international studies are taking place at the same institution than national ones.

In [4], we have examined typical conferencing and application sharing systems like Mbone [20] and NetMeeting [21] with respect to the catalogue of these requirements. It turned out, that none of them is fully covering the whole list. As a consequence, a new design has been made. To this end, we started with a suitable model.

D. *Model for Context Awareness in Teaching*

As mentioned already in the introduction, context awareness means the notification of persons about the state of one or more individuals including the state of all objects and elements in their environment. In general, this information is changing in time thus building a history. Moreover, persons and things can be subject to a predefined schedule. As a whole, the notion of context information comprises the evolution of states from past via presence to future and can be meaningful to other people to a more or less extent.

In teaching situations, context data are relevant from three different types of sources: sessions, participants and applications. A *session* is determined by the following information:

- History: past sessions being related somehow, progress of the current session in time and content, i. e. the overall results achieved so far.
- State of sessions being related to this session and taking place in parallel.

- Number, identification and state of each participant
- Number, identification and state of each shared application.
- Schedule.

Teaching sessions cannot be seen as isolated performances but depend on each other with regard to their content and participants. Each single part contributes to the overall learning process by delivering a subset of knowledge to mostly the same people. A session starts at some designated point in time and is making progress. Participants are discussing topics and presenting slides or experiments to mediate facts and relationships. To this end, they are using appropriate applications. Though not a regular case, the possibility exists that several sessions are taking place in parallel and have to be synchronized according to some predefined behavior. For instance, a large class can be split into small groups for an exercise and joined afterwards to continue plenary discussion. If they are mutually aware of their state, groups are getting the chance to adapt their speed and reduce the overall waiting time. In general, the sequence of treated topics can be predefined in a more or less strict way. Moreover, the session can be planned with a definite completion time or be open-ended.

Some relevant context information about *participants* of a session has been mentioned already in Section II.C. The complete list looks as follows:

- History: knowledge level and abilities
- Current state: Location, role, desire to ask/comment, degree of attentiveness, understanding, equipment and resources
- Schedule
- Privacy concerns

Teachers and students are motivated differently with regard to their interest in a certain participant. This is one example for the need of roles; additional ones are application specific as described in the next section.

Teachers have to take into account the history and current state of at least the majority of learners to be able to adapt their presentation adequately. This holds even for the equipment of learners due to the fact that the most wonderful multimedia slide is not worthwhile for people participating with small wearable devices (cf. [3]). Moreover, teachers should not overlook a student's intention to contribute actively, e.g. by raising his hand.

As described in the first scenario, finding teams of co-learners motivates the interest of students in each other. For them, knowledge level, location and schedule are the most relevant context information. But application specific roles can be of interest as well.

In either case, it must be totally up to each participant, whether to publish context information to everybody, to a subset of participants or not at all. If information is visible to others, it is useful in collocated scenarios already, especially in a large auditory. For remote participants, it plays an even more essential role. The most suitable way to present context information about participants is a map containing all participating persons with their position and additional information at appropriate level of detail.

For applications, only a few general statements are possible. Very often, more than one *application object* is involved. Similar to sessions and participants, such application objects should provide the following context information:

- History
- Current state
- Schedule
- Relations to other objects and to participants

Depending on the type of application, its objects either belong to the real or the electronic world or to both. As such, they are subject to a certain life cycle starting in the past and moving toward future in a planned or ad-hoc way. Objects can be interrelated in temporal, spatial or other fashion. Relations to participants define roles with regard to a certain object, i.e.. the right to modify or view its state. All these abstract considerations will be concretized when coming to examples in section V.

Before that, the realization of the model for context awareness in teaching is examined.

III. Architectural Framework of SASCIA

Basically, two alternate architectures are available for application sharing as a whole: one based on the client-server model and the other one on peers. When considering the lecture scenario taking place in a designated room it is probable to have at hand a computer and projector for presentations because more and more rooms at universities are going to be equipped with such devices. Hence, for this case it is obvious to have the server for application sharing running on this mostly powerful presentation computer whereas mobile devices with possibly less resources host the client part. Regarding however the scenario of encounters among students that can take place anywhere and maybe without any infrastructure at hand, a peer like fashion would be more adequate.

In a first step, we developed a client-server-based system by keeping in mind the general architecture of conferencing and application sharing systems. But the peer case will be tackled with in the near future. As shown in Fig. 1, clients and server are physically connected by a network, that can be of fixed or wireless nature. A suitable communication mechanism based on this network is providing logical connections between all components involved. We are assuming a channel-based metaphor where each participating component can register to receive those events that have been forwarded to the channel by others. By encapsulating the communication mechanism, different systems can be used and have to be specified for each instantiation.

We briefly introduce the functionality of each component. With a server running on some host, authenticated users can set up sessions via the *session administration* components. Inserting it into the session directory achieves public announcement of a session. Furthermore, context information as described in the last subsection can be maintained by this directory.

Via *participation* components, one can join a session either by selecting the appropriate session name in the session directory or by directly specifying the server and session name. Moreover, in combination with the *user directory*, these components care for proper authentication of participants and are storing context information about participants.

After preparations have been done, the actual teaching activities can take place. To this end, participants are using applications. Each of them is integrated into the system by instantiating and specializing a generic *application frame*. It consists of a number of components and provides functionality as follows:

- A configurable *floor control* component checks access rights of participants for input and control of the application.
- *Consistency control* cares for a semantic check of the feasibility of actions.
- Application specific *context* is captured and evaluated.
- It embeds the communication system, enables public and private *annotation* and provides *archiving* of all relevant data including context information into databases for later retrieval.

Usage of archived data can be internally or externally. For instance, recorded information can be needed by the component for context handling to derive missing data. External usage means retrieval by students, preferably after completion of a session..

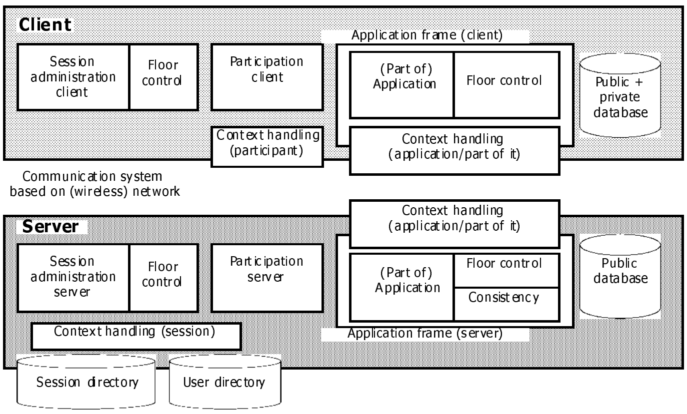

Fig. 1. Overall SASCIA architecture

In the following, we describe in more detail those components being essential for context awareness and consistency. A detailed treatment of the whole system can be found in [6].

A. Floor control based on context information

As a first mechanism to avoid chaotic behavior and care for consistency of control and data during a collaborative session, a component for floor control is needed. It is checking for proper authorization and access rights (so-called floor) of participants. Hence, an instantiation exists for the session management as well as for each application that is shared among participants.

Depending on the setting, participants are in a hierarchical or peer relationship. Different applications as well as these two settings are posing different requirements on *floor control*. To cope with the whole spectrum, a configurable component has been designed. It exhibits the following parameters:

- Floor size
- Sub-floors as possibility to partition the floor.
- Session style: moderated or voting to decide about granting or revoking the floor.

We introduce the size of a floor to be a number greater or equal to one. It stands for the number of participants that are granted the floor simultaneously. Thus, strict as well as relaxed conditions for sharing applications can be realized. In a lecture, rather the first case will be the one preferred because either the teacher or a student is allowed to perform annotations or control a tele-experiment. For a discussion among students, simultaneous usage of applications, annotation and pointing can be more advantageous to some extent.

If not set to one, the floor size should be specified as fraction of the group size to automatically adapt it if the number of participants changes. Otherwise two kinds of side effects might appear, either the unintended case of not letting everybody get the floor simultaneously in case of latecomers or the mostly uncritical one of multiple assignment of the floor to the same person after early leaves.

Thinking of simulations, experiments or games, participants can play different roles when using the same application, one for each application object. Hence, a partitioning of the whole floor makes sense to control input to each of these roles separately. This is comparable to the treatment in [12]. We proof the usefulness of this mechanism when describing example applications in Section V.

If a participant wants to perform an action and currently does not possess the floor or sub-floor, he has to request it. In a lecture setting, the teacher is informed about such requests and has to decide whom to grant the floor at first and how to proceed. As mentioned above, a map containing seat numbers of all participants and further characteristics like e. g. the person's knowledge and abilities can facilitate this decision.

In a setting with peers, granting the floor can be supported by voting. To this end, each participant is informed about queued floor requests and votes for each of them. The participant with the majority of votes gets the floor. In case of deadlock, either the team is informed about the situation or the system determines one of the candidates automatically. This behavior is subject to configuration.

B. Control of semantics based on context information

Even in case of input being in concordance to access rights of a participant, the semantics of this action can be faulty in the current context. Regarding applications like talking to a microphone, gesturing toward a camera or using an electronic whiteboard, faulty behavior can be annoying for other participants in the worst case. But when thinking of further applications, it can lead to serious problems yet. While learning by making errors and observing the effect of these errors is a very effective one for students, not all consequences can be tolerated.

In simulation environments being completely restricted to the electronic world, faulty usage is bearable as long as the availability of devices is not touched. But this does not apply to experiments in the real world. Taking examples like a railway model or chemical reactor, train crashes and explosions lead to damages and loss of expensive material. Hence, intended user actions have to be checked among one another and against the current context information to recognize erroneous input and prohibit it instead of causing such catastrophic effects. For instance, [24] apply a suitable collision detection to prevent robots from crashing into walls when being controlled remotely.

It can be concluded, that specific consistency rules have to be defined for each application. Only then, the input of participants can be checked automatically against these context depending rules and be accepted or rejected accordingly.

IV. Context Handling

As shown in the last sections, context information is crucial for application sharing in teaching environment. In the following, we examine mechanisms to get such information.

A. Context of participants

All kinds of context information about participants is managed in the user directory and thus available for further need. Different mechanisms are used to capture the information and transfer it into this directory. They subdivide into manual definition by persons and automatic derivation.

The *role* of a person is the only information being easily to determine. Whether a person is allowed to act as a teacher has to be marked when creating or modifying an entry for her in the user directory. If only authorized administration staff has write access to this directory, misuse should be excluded in general. Application specific roles are either assigned by a teacher or the group and inserted accordingly.

Further parameters of participants are more or less difficult to determine. In the meantime, a lot of systems have been built to find out about the *location* of a person. An overview is presented in [10] who classify according to criteria like cost as well as accuracy and precision that reach from some hundred meters down to millimeters. Due to high cost and effort of these systems, the SASCIA system currently relies on manual input of the seat number. The same applies to *hand raising* where the only alternative apparently being available consists in using cameras and image understanding.

[21] apply manual input for students' feedback to the teacher's presentation speed. This is one hint for *attentiveness* and *understanding* of students (further information regarding the level of concentration can be derived e. g. by the examination of eye movement as reported in [8]).

In contrast to these parameters, the *knowledge* level should not be specified manually due to the fact that self estimation is questionable. In a first approach, participants of an unguided experiment are classified by means of a short initial test. More extensive information can be collected by permanent control and observation.

B. Context of applications

By the interaction of people with an application, state changes of application objects are provoked, if they are allowed by control components for floor and semantic consistency. Thus, the application specific context comprising internal states and those of the environment is changed as well. To enable collaborative usage, participants must be informed about such changes where only a subset of them can be meaningful. Hence, events of state changes have to be captured, eventually evaluated and filtered as well as distributed. To make things more concrete, we briefly sketch the nature of states and corresponding changes for some popular applications in teaching: presentation tool and development environment. More examples are explained in the next section.

By means of *presentation* tools, a sequence of slides is shown. Thus, a state of this application contains the list of those slides being treated already as well as the number of the current one. With each slide, annotations of different type as appended during the performance are coupled closely. An annotation can be of written, oral or visual form. As a consequence, even drawings, keywords and gesturing for explanatory purposes are belonging to states of the presentation application. Moreover, synchronization points are of interest, e. g. switching to or returning from another application.

A *development environment* is responsible for the creation, modification and completion of specifications (e. g. UML diagram, interface declaration, program module) and relations among them. Hence, these specifications constitute application objects and their current state is one part of the state of the application as a whole.

Regarding *real experiments*, the chemical or physical state of all objects and elements involved are of relevance. Whereas states of all other applications as mentioned above are directly available in electronic form, those of real experiments have to be measured first and transferred to the electronic world.

V. Examples

A. Simulation of communication protocols

In the area of communication protocols for computer networks and distributed systems, students very often are confronted with rather complex, highly dynamic and concurrent processes. Think for instance of a protocol for the electronic market with buyer and seller trying to install a reliable and secure connection where they can trust each other. It is not obvious, how parameters can prevent a man-in-themiddle attack (cf. Fig. 2).

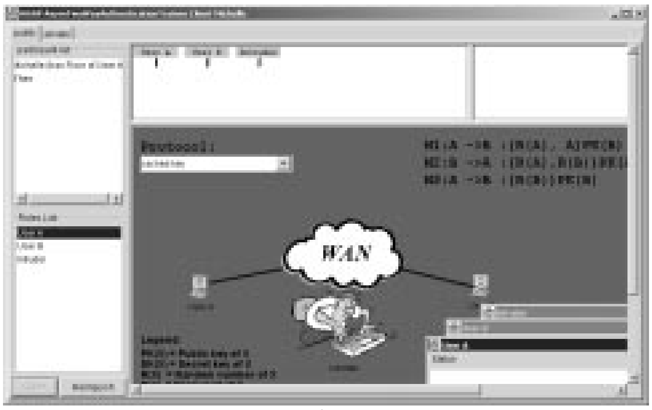

Fig. 2. Collaborative simulation of communication protocols

In [5], we have laid out a mechanism to facilitate teaching in this area. By means of highly interactive animation and simulation applets, one can experiment with communication protocols. The corresponding simulation environment consists of a number of communicating nodes, connections and messages. The state of a node comprises states of all its internal parameters; relevant events are sending and receiving messages of a certain type. At each point of time, a channel exhibits a certain bandwidth, throughput and error rate. Consequently, a message is transmitted correctly or with error states (e.g. lost, destroyed or multiplied).

In a first step, such applets enable a single learner to play with a protocol, try out different input sequences as well as correct and faulty protocol variants. Hence, she can watch the resulting behavior with respect to correctness and performance. Moreover, a teacher can demonstrate the whole process to students.

In a second step, collaborative usage of applets has been realized. Depending on the protocol in question, simulated nodes are behaving according to a certain role. For instance, in the mutual authentication scenario, three roles are involved (besides a moderator): buyer, seller and attacker. The floor control as described above cares for offering a sub-floor for each of these roles, thus enabling participants to play together in a controlled manner.

It should be noted, that these simulation applets have been constructed according to the MVC-scheme (Model, View, Control). This enables the afore mentioned possibility to partition applications. Whereas the model and control parts are residing on the server side, each client has only a view part running locally. After each simulation step, the server notifies clients about the new context information. Hence, the component for view can provide context awareness to participants by showing the current state of each simulated object.

B. *Control of three tanks*

The following presents a combined computer exercise and laboratory experiment in the area of control engineering, that will be set up for a lecture named *LMIs (Linear Matrix Inequalities) in Control*. The experimental setup consists of three tanks filled with blue water (see Fig. 3). The height of each water column (h_1, h_2, h_3) is measured using piezo-sensors setup. The water flow into the first and the third tank can be set to any value between 0 and the maximal flow by an electric signal to the respective valve. The outflow depends on the height of the water column in the third tank.

In the described experiment, the objective is to control the level of tank three. Neither the levels of the tanks one and two nor the additional disturbance inflow $d(t)$ in tank three are known. A controller has to be designed using the information on the height in tank three to set the inflow in such a way that the height h_3 tracks a reference height y_{ref}. This tracking should be as good as possible where good is mathematically defined as limiting the effect of the disturbance flow on the tracking error $h_3 - y_{ref}$ (in the H_2 sense) and the effect of the reference y_{ref} on the tracking error (in the H_∞ sense).

The first part of the students' task is to model the system, design a controller, and optimize the parameters to get the best performance. Several pieces of context information characterize the students' *progress*. For example, one could characterize which steps the respective students have completed, which of these are correct, what their controller structure is and what controller parameters they have chosen.

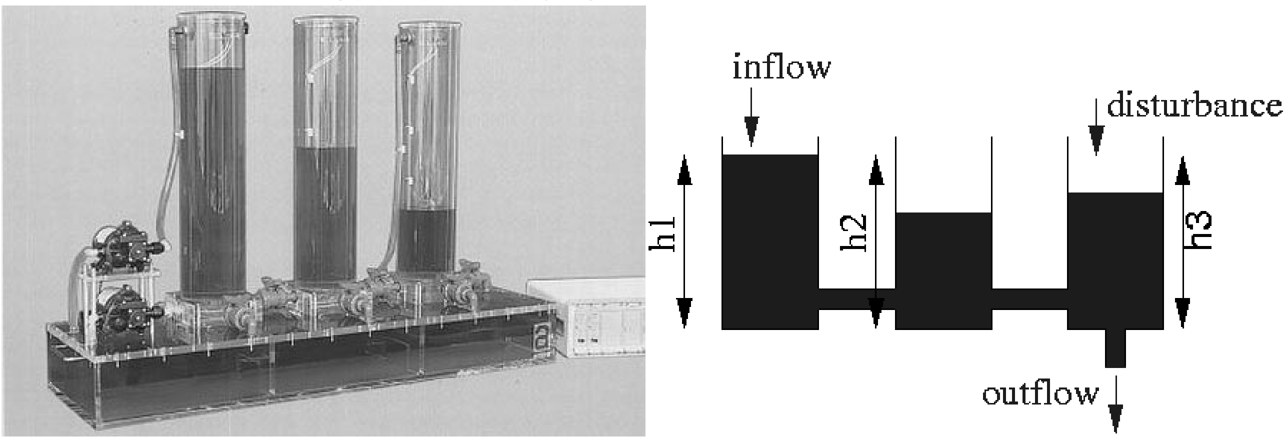

Fig. 3. Three-tank experiment.

Also the achieved control performance measures on the experimental setup and in simulation can be made accessible to the supervisor or to other students (e. g. as a list of best results).

The context information of the simulation experiment consists of the water *levels* in each tank, the reference level y_{ref}, the input *flow* u and the disturbance inflow. All these parameters depend on time. Only h_3 and y_{ref} are public and u can be set to any flow achievable by the pump. But for the visualization, all pieces of information are accessible. The reference and the disturbance can either be fixed a-priori to get repeatable experiments, or another student or the supervisor can set these parameters (resulting in a two or even three player game).

Besides the simulation, the whole experiment can be performed in real life manner. In this case, the context information is the same as for the simulation. The software-based visualization is replaced by a web-camera, as the private information is not accessible on-line. Furthermore, extra context information is necessary, especially whether the experiment is free or who is using it for testing a controller.

C. Control of a railway model

In this section, the importance of context awareness is shown by means of a tele-experiment based on a railway model. The didactical challenge is, in short, to avoid boring situations for the students but keep them motivated and listening. They can perform this exercise location independent and control the railway model from remote. Nevertheless, the tele-experiment should provide the same impression to students than the local one. Hence, hard requirements are posed on context awareness for this setting.

The railway model is a typical example for a complex technical system. Teaching issues concern the demonstration of contiguities and principles used for the control of automation systems. Examples are scheduling and synchronisation mechanisms or development of control software with Petri nets and logical blocks. Although the behaviour of a railway model is deterministic within a discrete schedule (compared to the three-tank-experiment of the last subsection), the complexity of the automation system depends on the size of the model.

As shown in Fig. 4 and 5, our test environment is consisting of two floors (one sub-floor underneath the table, not to be mixed up with access rights as described in Section III) with 4 stations possessing between 2 and 5 tracks, a freight station with a crane loading installation and an engine shed with locomotive depot and a transfer table. More than 10 separate trains can be run on the model. Hence, experiments at different levels of complexity can be specified.

Depending on the knowledge level and progress of participants, collaborative execution of such experiments can easily lead to unforeseeable and chaotic situations. If no teacher is present to prevent damage, the system itself needs suitable mechanisms. To this end, two different aspects of context awareness have to be considered at the beginning: The behaviour and state of the participants who are executing the experiment as well as the performance and state of the technical process, i. e. the railway model.

Attending students exhibit different prerequisites regarding *knowledge* and desired *execution form*. Some participants prefer a guided tour through the experiment, others want to solve their tasks independently. The behaviour of a single participant has to be considered as well as those of all participants attending the experiment at the same time. All these factors have to be taken into account when deciding about those experiments being available from remote.

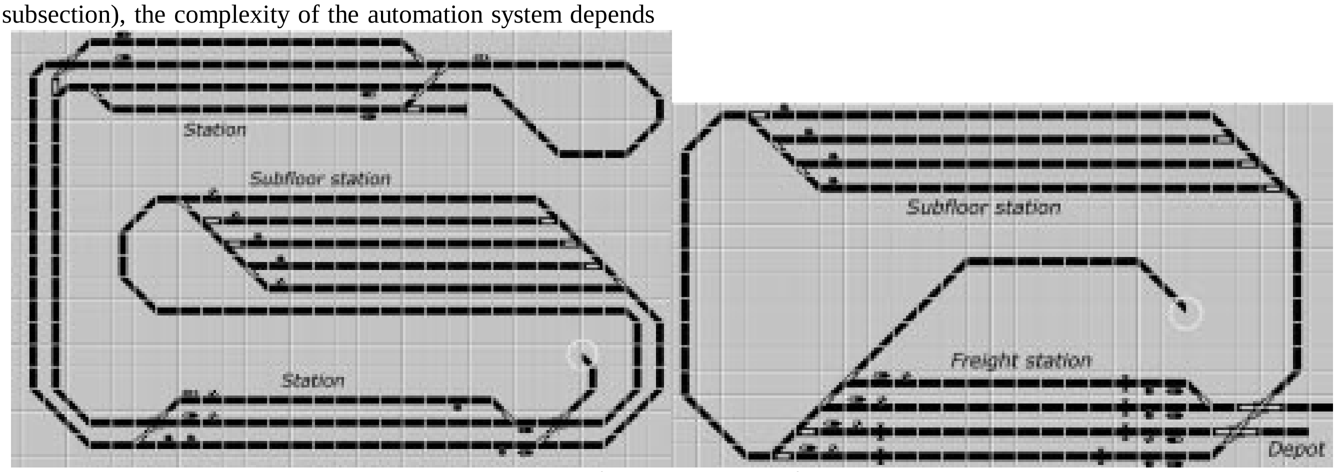

Fig. 4. Track view of the two track circuit of the railway model

Fig. 5: Railway model

For the control of a technical process from remote, a lot of context information is needed. In the case of the railway model, digital *signals* can be gathered to get information e. g. on the allocation of *tracks*, *switch* stands, light and semaphore signal states, *train* positions and movement, *crane* positions, position of the *transfer table*. The collection of this information spans the state space of the railway model. Due to its size and complexity, it is not manageable by persons from remote. There are two complementary means to deal with this problem. Firstly, the whole set of information has to be prepared and delivered to participants in a concise manner. Secondly, the possibilities for remote control can be limited to a certain area (e. g. a station or an engine shed) and to a number of predefined actions executable from remote, according to participants' abilities.

During the actual execution of the experiment, two kinds of consistency have to be ensured: a) the consistency of single actions and b) the consistency of execution with reference to the real world railway model. At first, simulations are used to check and validate the students' results before really executing them on the railway model. Thus, students can get an impression of the effects of their results without endangering the security of the experiment.

It should be mentioned, that Web cameras are delivering additional visual data, but serve as auxiliary medium only due to delays and view limitations. Cameras provide feedback about the success of operations but are not suited for real-time control.

VI. Project State and First Results

The overall framework of SASCIA containing user and session management as well as generic application frames has been implemented. The following applications are available for collaborative usage so far: chat facility, whiteboard for annotation and simulation of communication protocols.

Before testing SASCIA in a real class, a series of measurements has been performed. We estimated the most critical parameters to be the time for storing and retrieving data to/from the database that is used to manage context information about applications. To this end, different variants have been tested. The encapsulated database InstantDB [15] is offering two modi, one with storing directly to disk ('without fastUpdate') and another one with keeping data in memory and writing them to disk subsequently ('fastUpdate'). Fig. 6 demonstrates that this factor does not provide good results. For better performance, a further buffer was introduced to receive data before passing them to the database. But even this variant does not perform particularly well when compared to a solution solely relying on memory without any database at all. Nevertheless, values for the variant with buffer are acceptable.

VII. Conclusion and Outlook

We have presented a systematic approach to integrate context awareness into systems for application sharing in the teaching area. The resulting framework is open to include further applications as well as new services to capture and evaluate context data. The prototypical realization is

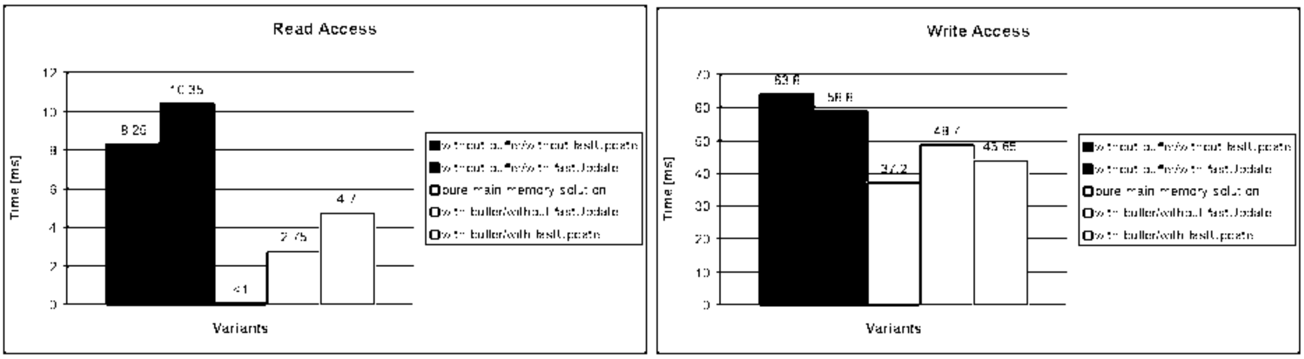

Fig. 5. Access time for reading and writing context information

containing some first applications and is performing quite well. The most remarkable of these applications is a simulation environment for collaborative exploration of communication protocols.

It is planned to integrate further applications like more simulation environments and real life engineering experiments, especially the three-tank- and railway-exercises as described in Section V. While the current implementation is focused on collocated participants, further plans concern the extension of the system to integrate the remote case and small devices like PDAs. Moreover, an architecture based on peers and a mechanism for the synchronization of parallel sessions will be examined.

As a whole, SASCIA will contribute to better interactivity among teachers and students as well as increase the efficiency of teaching.

ACKNOWLEDGMENT

Our special gratitude is to the leader of the distributed systems group, Kurt Rothermel for fruitful discussions and to the team of those students that have designed and implemented the SASCIA framework: Peter Oberparleiter, Xue Bai, Andreas Schmid and Marcus Sommer.

REFERENCES

[1] M. Bauer, G. Kortuem, Z. Segall. Where Are You Pointing At? A Study of Remote Collaboration in a Wearable Video-Conference System. Proceedings Third International Symposium on Wearable Computers (ISWC'99) , 18-19 October, 1999, San Francisco, California.

[2] R. Bentley, T. Horstmann, J. Trevor. The World Wide Web as Enabling Technology for CSCW: The Case of BSCW. Computer Supported Cooperative Work: The Journal of Collaborative Computing 6: 111-134, 1997

[3] O. Brand, W. Mahalek, D. Sturzebecher, M. Zitterbart. MACS - A Modular Collaboration Environment. Proc. IASTED International Conference on Internet and Multimedia Systems and Applications (IMSA '99), Nassau, Bahamas, October 1999

[4] E. Bullinger, C. Burger, S. Papakosta and T. Wagner. Investigation of application sharing systems for teaching purposes in engineering disciplines. Submitted to 4th International Conference on New Educational Environments

[5] C. Burger and K. Rothermel. A framework to support teaching in the area of distributed systems. *acm Journal on Educational Resources in Computing (JERIC)*, Volume 1 (2001), available at http://www.acm.org/pubs/contents/journals/jeric/2001-1/

[6] C. Burger, S. Papakosta and K. Rothermel. Application sharing in teaching context with wireless networks. Accepted for World Congress on NETWORKED LEARNING IN A GLOBAL ENVIRONMENT 2002

[7] A. Eckert et al.. A Distance Learning System for Higher Education Based on Telecommunications and Multimedia. Proceedings ED-MEDIA/ED-TELECOM'97, Calgary, June 1997.

[8] L. Garber. Polish Students Win Computer Society Design Competition. IEEE Computer, August 2001, Vol. 34, No. 8, pp. 67-71

[9] M. Guggisberg, P. Fornaro, T. Gyalog and H. Burkhart, An Interdisciplinary Virtual Laboratory on Nanoscience , Electronic Notes in Future Generation Computer Systems, Vol. 1 (2001). Also available at http://www.nanoworld.unibas.ch/zope/nano/articles/proceedings_speedup

[10] J. Hightower, G. Borriello. Location Systems for Ubiquitous Computing. IEEE Computer, August 2001, Vol. 34, No. 8, pp. 57-66

[11] A. Hills. "Wireless Andrew", IEEE Spectrum, Vol. 36, No. 6., June 1999

[12] V. Hilt, W. Geyer. A Model for Collaborative Services in Distributed Learning Environments. Proc. International Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services 1997 (IDMS'97), Darmstadt, Germany, R. Steinmetz, L. Wolf (Eds.), LNCS 1309, Springer Verlag, Berlin, Germany, pp. 364-375, 1997

[13] F. Hohl, U. Kubach, A. Leonhardi, K. Rothermel, M. Schwehm. Next Century Challenges: Nexus - An Open Global Infrastructure for Spatial-Aware Applications. Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, Washington, USA, August 15-20, 1999, T. Imielinski, M. Steenstrup, (Eds.), ACM Press, 1999, pp. 249-255

[14] A. Hopp, D. Schulz, W. Burgard, A. B. Cremers, D. Fellner. Virtual reality visualization of distributed tele-experiments. In Proc. of the 24th Annual Conference of the IEEE Industrial Electronics Society (IECON), 1998. Also available at http://www.informatik.uni-bonn.de/~schulz/

[15] InstantDB. http://www.lutris.com/index.html

[16] JETS 2000. A Java-Enabled TeleCollaboration System. http://www.mcrlab.uottawa.ca/jets/

[17] P. Ljungstrand. Context-awareness in distributed communication systems. Workshop "The What, Who, Where, When, Why and How of Context-Awareness", Computer-Human Interaction (CHI) 2000, The Hague, The Netherlands, 2000. Also available at http://www.viktoria.se/play/publications/2000.html

[18] K. Lund et al.. Requirements Analysis and Design for a Flexible Learning-on-Demand System. Proc. Integrated Design & Process Technology, Dallas, June 2000

[19] R. Malpani, L.A. Rowe. Floor Control for Large-Scale MBone Seminars. Proc. of The Fifth Annual ACM Intl. Multimedia Conf., November 1997. http://www-mash.cs.berkeley.edu/frame/research/publications/

[20] MBone. http://bzvd.urz.tu-dresden.de/mbone/software.html

[21] mlb http://www.informatik.uni-mannheim.de/informatik/pi4/projects/mlb/

[22] NetMeeting. http://www.microsoft.com/windows/netmeeting/

[23] Pebbles. http://www.cs.cmu.edu/~pebbles/classroom/index.html

[24] A. Schmidt, M. Beigl, H.-W. Gellersen. There is more to Context than Location. Environment Sensing Technologies for Adaptive Mobile User Interfaces. International Workshop on Interactive Applications of Mobile Computing (IMC 98), November 1998, Rostock, Germany.