Group-based Learning Using a Remote Laboratory

Christof Röhrig and Andreas Jochheim
Department of Electrical Engineering
University of Hagen
D-58084 Hagen, Germany
christof.roehrig@fernuni-hagen.de

Abstract

This paper presents a collaborative learning environment for a remote laboratory. Students have access to the remote laboratory via Internet from anywhere at any time. They control the experiments exclusively with their standard Web browser, no additional software is needed. The learning environment allows the experimentation in a team. The group is able to interact and to discuss the results of their work. A real collaboration like in local experimentation is possible. The remote laboratory is based on a client/server architecture, which is mainly implemented in the Java programming language. Furthermore the contribution discusses the motivation of group-based learning using remote experiments.

1 Introduction

In distance education, laboratory experimentation is inconvenient because the students usually have to be physically present in the universities laboratories. A solution to avoid the disadvantages of local experimentation is teleoperation of laboratory experiments. Providing remotely accessible experiments, unique or expensive equipment can be shared between several universities. So, a larger number of laboratory resources is available, and students can choose from a variety of laboratory experiments.
In local laboratory experimentation students usually work together in groups of two or more. This learning paradigm is often called collaborative learning. Collaborative learning develops skills for solving problems in a team. The underlying premise of collaborative learning is based upon consensus building through cooperation by group members, in contrast to competition in which individuals best other group members. Learning members of the group will usually organize their activities themselves and decide upon the roles of the different members via consultation and negotiation. [1]
With the rapid expansion and availability of communication and information technologies, collaborative learning can also be done effectively in a virtual environment at a distance. Collaborative virtual environments bring together users, which are geographically distributed, but connected via a network.

2 Synchronous Communication Techniques

To provide remote experimentation to a group of students, synchronous communication techniques are required. In conjunction with the Internet different communication techniques are common:
Text-chat: In a text-chat the users exchange textual messages by typing in a shared space. Communication in larger groups are difficult because it gets very hard to follow the train of thought. Text-chat is suitable even with very slow Internet connections.
Audio-conference: In an audio-conference the voice is captured and transmitted over Internet to other participants. The user needs a sound-card with additional microphone. The bandwidth requirement is much higher than in a text-chat.
Video-conference: In video-conferences the additional video is used to watch other participants and to be aware of their existence and actions. A video-conference has much more bandwidth requirement than a audio-conference. If a true collaboration of all partners is desired, the partner with the smallest bandwidth limits the communication. It is not suitable for users at home with slow Internet connections.
3D-chat: In 3D-chat an *avatar* tries to mimic the behavior of the user in virtual reality. The *avatar* plays the role of the video in a video-conference. Participants can see other users as in real world. 3D-chats have lesser bandwidth requirements than video-conferences, because only events are transmitted. 3D-chat can be combined with text-chat or audio-conferencing.

In distance education at home video-conferencing is not suitable because of bandwidth limitations. The remote laboratory requires real interaction between the students and the tutor, so a bandwidth-saving way of interaction is required as an alternative to a video-based communication. Pure text-chat does not meet our requirements because a multi-user remote laboratory application needs the possibility of real interaction. The tutor has to be enabled to introduce and to explain the details of the experiment by some kind of visual representation of the experiment. A 3D-chat in conjunction with text-chat for synchronous communication and collaboration was chosen. The technology behind a collaborative learning environment is a mechanism that enables a user to send up-

dates to other users about the interactions that are made in the shared environment. It is necessary for the participants to have the same view of the application in real time [2]. A comparative summary of the different standards for collaborative environments is given in [3].

3 System Design

The system design is based on a client/server architecture written mainly in the Java programming language. Students may work on any platform that supports a Web browser with a Java runtime environment. Java is used to eliminate the operating system problem of heterogeneous environments, such that users are not restricted in their choice of a resource. This is specially important for distance education since some users might choose UNIX-workstations, while others might prefer Windows 95/98/NT PCs or MACs. The introduction of Java helps to overcome these problems. The local Web browser is the only user interface to the experiment. The browser loads the client software as Java applets from the server and starts them. The collaborative environment is divided into two main modules: a rendering and graphics part on the client side and a communication middleware on the server side. On the client side VRML is used to display the virtual 3D environment. VRML as a text-based language is a powerful, nevertheless simple language to build 'virtual worlds', which include 3D objects, light sources and animations.

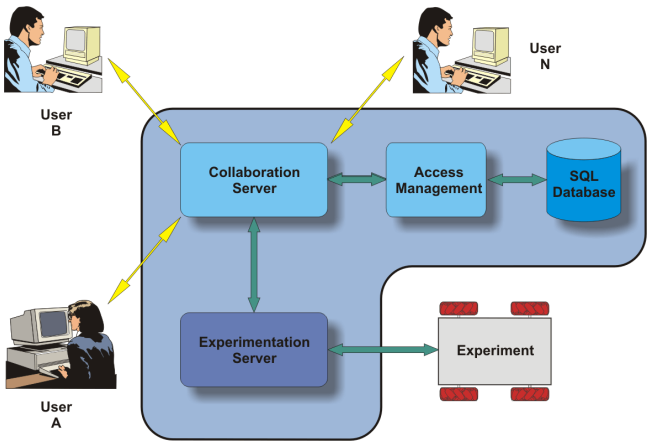

Figure 1: Communication Structure

Figure 1 shows the communication structure of the collaborative environment. The communication middleware includes four servers. The first server is the Web server which provides the HTML documents, the VRML scene, VRML avatars and all Java applets. The second server is the collaboration server to which the clients establishes connections in order to do the experimentation and to telecollaborate. The third server is the experimentation server which send commands to and receives status information from the experiment. The forth server is the access management system which does the scheduling of experimentation time. Appointments are stored in a SQL database. The user interface of the access management system is implemented in Java and executed in the Java runtime environment of a Web browser. More details of the access management system are described in [4]. Technical details of the collaborative virtual environment are described in [5].

4 Conclusion

The contribution shows that distance education can be applied to real laboratory experiments. Even collaboration in a team is possible. Remote laboratory users are able to interact and discuss the results of their work. A real collaboration like in local experimentation is possible. On the client side, there are only some minor requirements. Students are able to use the experiments with a Web browser and Java runtime environment. In the laboratory, the experiment must be adapted to the requirements of remote control. All software developed for the laboratory has been implemented in the programming language Java. Therefore, it can be easily adapted to different platforms.

References

[1] P. Kirscher, "Using Integrated Electronic Environments for Collaborative Teaching/Learning," in *Proceedings of the 8th Annual Conference of the European Association for Research on Learning and Instruction*, Göteburg, Sweden, Aug. 1999.

[2] S. Shirmohammadi, J.C. Oliveira, and N.D. Georganas, "Applet-Based Telecollaboration: A Network-centric Approach," *IEEE Multimedia*, vol. 5, no. 2, pp. 64–73, 1998.

[3] J.C. Oliveira, S. Shirmohammadi, and N.D. Georganas, "Distributed Virtual Environment Standards: A Performance Evaluation," in *Proceedings of the 3th IEEE/ACM International Workshop on Distributed Interactive Simulation and Real Time Applications*, Greenbelt, USA, Oct. 1999.

[4] C. Röhrig and A. Jochheim, "Java-based Framework for Remote Access to Laboratory Experiments," in *Proceedings of the IFAC/IEEE Symposium on Advances in Control Education*, Gold Coast, Australia, Dec. 2000.

[5] A. Bischoff and C. Röhrig, "Remote Experimentation in a Collaborative Virtual Environment," in *Proceedings of the 20th World Conference on Open Learning and Distance Education*, Düsseldorf, Germany, Apr. 2001.