The Digital Lecture Board - A Teaching and Learning Tool for Remote Instruction in Higher Education[1]

Werner Geyer, Wolfgang Effelsberg
Lehrstuhl für Praktische Informatik IV,
University of Mannheim, Germany
{geyer,effelsberg}@pi4.informatik.uni-mannheim.de

Abstract: This paper presents a novel, integrated teaching and learning tool - called digital lecture board - which takes into account the requirements of synchronous, computer-based distance education. For almost two years, the TeleTeaching project Mannheim-Heidelberg has been using video conferencing tools for transmitting lectures and seminars. These tools prove to be insufficient for the purpose of teleteaching since they are not powerful enough to support team work, they are not flexible enough for the use of media, and are somewhat difficult to handle by non-experts. We discuss shortcomings of the existing tools and disclose features we had in mind while designing the digital lecture board. Embedded in a teaching and learning system, the digital lecture board even allows for asynchronous usage modes, for instance, the preparation of lectures. Moreover, we cover implementation issues of the current prototype.

1 Introduction

The advent of powerful hardware and advances in high speed networks enabled synchronous learning, with teachers and students being geographically distributed but connected via computer networks. Compared to traditional distance learning, this allows for a higher degree of interactivity and collaboration but still entails crucial drawbacks compared to the traditional classroom situation. Today's synchronous learning systems are mainly based on video conferencing technology which provides audio, video, and joint editing of documents only, but does not take into account the specific requirements of teaching, for instance, controlling the course of instruction, raising hands, or reference pointing. A shared whiteboard for transmitting slides is often the core part of these systems but mostly also the bottleneck, since the teacher is forced to tailor his or her course to the limited features of the whiteboard software. The *digital lecture board* presented in this paper is a novel, integrated teaching and learning tool which is basically an extended whiteboard tailored to the needs of synchronous teleteaching.

2 Background

The development of the digital lecture board is based on the experience we gathered in the TeleTeaching project Mannheim-Heidelberg. The project aims at an improvement in the quality and the quantity of lectures by using multimedia technology and networks for the distribution of lectures and seminars. We have implemented three different instructional settings - as indicated in Figure 1 - which are characterized by their scope of distribution, interactivity, and individualization of the learning process.

In the *Remote Lecture Room* (RLR) scenario, audio/video-equipped lecture rooms are connected via a high speed network, and courses are exchanged synchronously and interactively between participating universities. *Remote Interactive Seminars* (RIS) are a more interactive type of instruction where small groups of participants are distributed across seminar rooms connected by a network. RIS focuses mainly on the co-operative, on-line construction and presentation of reports. The *Interactive Home Learning* (IHL) copes with maximal

[1] This paper has been submitted to ED-MEDIA'98, World Conference on Educational Multimedia and Hypermedia, Freiburg, Germany, June 1998.

geographical distribution of all class participants: Each student learns asynchronously as well as synchronously at home in front of his/her PC. Obviously, the IHL scenario makes the greatest demands on technology (e.g. low-bandwidth transmission of audio and video) and pedagogy (e.g. controlling the course of instruction and/or human interaction). For a thorough description of pedagogical, organizational, and technical issues of the project refer to [Eckert et al. 97], [Geyer et al. 97].

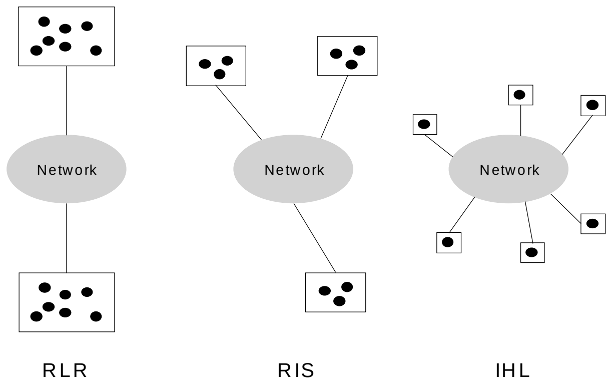

Fig. 1: Instructional settings in the TeleTeaching project Mannheim-Heidelberg.

So far, we have been using the Internet and the MBone video conferencing tools *vic* (video conferencing tool), *vat* (visual audio tool), and *wb* (whiteboard) for remote lecturing [Macedonia & Brutzman 94]. Observations, surveys and interviews with the students and lecturers during the last two years indicate that these tools can provide satisfactory results if the lecturer adapts the layout of the lecture exactly to the limited features of these tools, but are far from optimal for teleteaching since they have not been designed for this purpose. This concerns specifically the whiteboard, which can be considered to be a substitute for the traditional blackboard. Along with audio, the whiteboard is most important for conveying knowledge to distributed participants. In order to overcome the weaknesses of the whiteboard, we decided to develop the *digital lecture board (dlb)* which will better satisfy the needs of the instructional settings RLR, RIS, and IHL.

3 Design Considerations

In this Chapter, we present, in more detail, the shortcomings of the existing MBone tools[2], and we discuss the most important features which we had in mind while designing the dlb.

3.1 Integrated User Interface

The MBone tools do not provide an integrated user interface. Teachers and students complained about many confusing windows and control panels which are not important for learning and teaching but make it more difficult to operate the tools. Since teleteaching at the university level should not be restricted to computer experts, we find it especially important that the dlb provides an easy-to-operate user interface which integrates also audio and video communication. Moreover, in order to allow the interface to adapt to different instructional settings, it should be configurable. RLR, for instance, mainly focuses on presentation of knowledge to passive receivers and, therefore, receivers do not want to see tool or page selecting options on their screens.

3.2 Media Usage and Handling

[2] The described shortcomings are not limited to the MBone video conferencing tools but concern also other systems like NetMeeting, CUSeeMe, ProShare, etc.

The MBone whiteboard wb is very limited concerning the use and handling of media: only postscript and plain ASCII text are supported as external input formats, and later editing (besides move, copy and delete) of the built-in graphic and text objects is not possible. In addition, wb does not support joint editing of these objects. For instance it is impossible for a distributed group to create a common text, or to modify objects created by different participants. Since media are very important for a modern instruction, the dlb should support a variety of media formats such as GIF, TIFF, JPEG, MPEG, AVI, Postscript, HTML, AIFF etc. as well as many built-in object types such as lines, rectangles, circles, freehand drawings, polylines, text etc. Technically, objects are required to be editable by every participant and the dlb should provide functions like select, cut, copy, paste, group, raise, lower etc. similar to a word or graphic processing software.

3.3 Workspace Paradigm

The shared workspace of wb is limited to a two-layer concept with a postscript slide in the background and drawings and text in the foreground. It is, for instance, not possible to render two different postscript slides onto a single page so that results of two distributed work groups may be compared. Moreover, wb adheres to the strict WYSIWIS (What You See Is What I See) paradigm. Thus, participants cannot have a private workspace where they can prepare materials, for instance, when doing on-line group work. Teleteaching software requires a more flexible workspace concept with multiple layers where arbitrary media objects (audio, video, images, animations, etc.) can be displayed, grouped, raised, lowered, etc. Single participants or small groups should be offered private workspaces (invisible to the rest of the whole group) in order to allow for modern types of instruction such as group work. The outcome of the group work can be transferred to the shared workspace so as to allow a wider discussion of the results.

3.4 Collaborative Services

Today's teleteaching systems suffer a lack of communication channels compared to the traditional face-to-face instruction, since most systems support only audio, video, and joint editing of documents. Social protocols or rules control the human interaction and the course of instruction in a classroom. These mechanisms are difficult to reproduce in a remote situation and include, for instance, raising hands, giving rights to talk or to write on the blackboard, setting up work groups, and reference pointing. Collaborative services provide mechanisms to support the communication of persons through computers and to increase social awareness. Basic services such as floor control, session control, telepointers, or polling should be supported by the dlb. A detailed analysis of collaborative requirements in teleteaching indicates that the following situations should be supported by computer-based teaching and learning systems [Hilt & Geyer 97]:

- Joining a session at the beginning or later
- Leaving a session at the end or earlier
- Removing a participant from the session (e.g. by the moderator)
- Raising hands (signaling)
- Drawing back signals
- Removing signals by the moderator or teacher
- Selecting participants (with and without signal)
- Selective granting and removing of permissions (floors)
- Returning own permissions
- Pointing to shared instructional materials
- Creating, joining, and closing sub-groups (for group work)
- Private discussions and co-operation
- Nominating moderators for sub-groups
- Surveying sub-groups and participants (super user role of the moderator)
- Granting of permissions (floors) for work areas and materials to sub-groups
- Accessing work areas and materials without a moderator in order to enable internal group discussions

It is interesting to note that most of the computer-based tools for remote instruction are ignoring these types of interaction completely. In the dlb project, we attempt to build, and experiment with, electronic surrogates for social protocols.

3.5 Synchronized Recording and Playback of Sessions

The dlb should also provide the possibility to record a transmitted lecture or course which can then be stored on a server for playback. Students will then be able to retrieve the lecture in order to review certain topics or the complete lecture if they have missed it. Recording should include all transmitted data streams, i.e. audio, video, whiteboard actions & media, telepointers, etc. In order to achieve a synchronized recording, data has to be time-stamped. The data streams could then be recorded by existing systems like the VCRoD service (Video Conference Recording on Demand) [Holfelder 97] or the mMOD system (multicast Media-on-Demand) [Parnes 97]. These systems rely on the Real-Time Transport Protocol RTP for synchronized recording [Schulzrinne et al. 96]. The current release of the MBone whiteboard wb does not support the RTP standard.

3.6 Storage and Retrieval of Pages and Teaching Materials

Lectures or courses given with the computer need to be prepared in advance, i.e. producing slides, images, animations, etc. The preparation of materials with the MBone whiteboard is limited to a list of postscript files which can be imported by mouse click during a running session. In order to allow for a better preparation of on-line lectures and for saving results after a lecture, the dlb should support storage and retrieval of pages and objects in a structured, standardized file format such as SGML. Dlb documents would then be readable and modifiable by the human user or by other programs (e.g. a viewer software for lectures). Moreover, it would also be desirable for the dlb to have access to a multimedia database which stores teaching and learning materials of teachers and students. Such databases are also a research issue in our TeleTeaching project [Eckert et al 97].

3.7 Network Transmission

Scaleable and efficient transmission of data to a group of participants is only possible using multicast. Since we rely on the Internet for remote lecturing, we need to employ the Multicast Backbone (MBone) for multicast data delivery, as do the MBone tools vic, vat, and wb. Basically, multicast IP is an unreliable, packet-oriented protocol. In contrast to audio and video data, lecturing materials need to be transmitted reliably. Following the ALF concept (Application Level Framing), wb implements the reliability protocol within the application [Floyd et al. 95]. We think that separating the reliability issue from the application is a more appropriate approach for our dlb since this eases implementation and also allows other applications (e.g. a collaborative services module) to use reliable transport services.

Another important issue is security. We believe the dlb should support encryption in order to allow private sessions and billing. Prior to joining a specific course, students would have to register in order to receive an access key.

4 Usage Modes in a Teaching and Learning Environment

A teaching and learning environment consisting of a full-featured dlb (according to Chapter 3) along with a recording facility (e.g. the VCRoD service), a multimedia database for teaching materials, a WWW server, a VCRoD editing tool, and a WWW dlb document viewer would allow for several synchronous and asynchronous usage modes (see Fig. 2).

1. **Preparation/Pre-Authoring (asynchronous)**: Teachers and students use the dlb for the preparation of material for the synchronous teleteaching mode. They may access a multimedia database (MDB) or a local storage device (ls) for the retrieval of multimedia material needed for teaching and learning. The outcome of this mode is an off-line *dlb document* which can be, for instance, a complete presentation for RLR, a small report prepared by students for RIS, or a piece of group work for IHL. The dlb document can be stored locally (ls) or on the multimedia database (MDB).
2. **Transmission/Teleteaching (synchronous)**: The prepared dlb documents are used as a basis for synchronous teleteaching in the three instructional setting RLR, RIS, and IHL. The material is transmitted to the entire class. Teachers and students can then employ the advanced synchronous features of the dlb such as, for instance, reference pointing (telepointer), forming sub-groups (session control), annotating (drawing tools), controlling the course of instruction (floor control), on-the-fly development and import of materials (drawing tools and access to ls or MDB), or discussions (audio and video). The result of this mode is on the one hand a *modified dlb document* which can be stored by both teachers and students in order to save results. On the other hand, the VCRoD service allows for the recording of the complete teleteaching session including all media streams (audio, video, telepointers, dlb actions, etc.).
3. **Revision/Post-Authoring (asynchronous)**: Based on the materials obtained from the transmission mode (*dlb document* and *VCRoD recording*) , a complete multimedia document, enriched with further multimedia components (e.g. animations), can be produced by using the dlb, a VCRoD editor or further authoring tools. The multimedia document can be distributed off-line by CD-ROM or on-line via WWW. Obviously, the advantage of this approach for authoring is that production time is considerably reduced since material obtained from usage mode 1 and 2 is basically a spin-off of synchronous teleteaching[3]. In addition to the publishing issue, students may also use the *modified dlb documents* of usage mode 2 in order to produce individualized, annotated lecture notes with the dlb, e.g. for the preparation of exams.
4. **Retrieval (asynchronous)**: A light-weight WWW dlb viewer can be employed for retrieving lecture notes (dlb documents) from a WWW server for the purpose of viewing and printing. Moreover, the VCRoD service offers students who missed the lecture or who want to review a certain difficult topic the possibility to playback recorded lectures (*VCRoD recording*) as originally captured. The VCRoD service supports random access, fast-forward and rewind to arbitrary parts of the lecture [Holfelder 97].

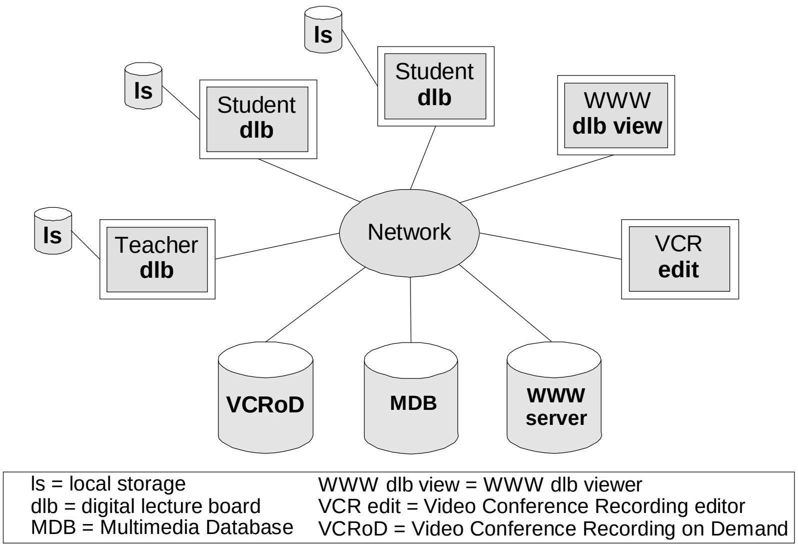

Fig. 2: Teaching and Learning Environment.

5 Implementation Issues

3 This procedure is very similar to the „Authoring on the Fly“ concept but differs in that our approach does not limit the recording to the teacher’s actions [Bacher & Ottmann 96].

5.1 Status

We have implemented a prototype of the dlb which fulfills most of the requirements mentioned in Chapter 3. Since the source code for the MBone whiteboard wb is not publicly available, we had to develop the dlb from scratch. To allow for a high degree of portability, we implemented the prototype in C++ and the Tcl/Tk scripting language [Ousterhout 94, Welch 97] and we took great care to reuse only components which are available on all major hardware/software platforms (e.g. ghostscript for rendering postscript pages). The current version has been successfully installed on the Unix systems AIX (IBM), IRIX (SGI), Solaris (SUN), and Linux (PC). The following features are supported:

- reliable multicast/unicast transmission using smp (scaleable multicast protocol)
- late join (students joining the session too late receive all materials up to date)
- media formats: postscript, gif, ppm, ascii
- graphical objects: freehand lines, lines, arrows, polylines, polygons, rectangles, ovals
- editable text objects with different font types, styles, and sizes
- joint editing of graphical objects and text (source synchronized)
- clipboard feature: cut, copy, paste
- zooming
- on-line and off-line mode (provides a private workspace)
- SGML-like document format (allows storage and retrieval of *dlb documents*)
- partly configurable, easy-to-operate user interface
- collaborative services (floor control, session control)
- telepointers
- RTP standard compliant network protocol compatible with the AOF whiteboard [Bacher & Ottmann 96]

5.1 Architecture

The dlb system comprises several components as indicated in Figure 3. On the *application level*, the core part of the dlb embeds functional modules which are basically encapsulated C++ classes. The postscript module (*ps*), for instance, supports rendering of postscript pages by interfacing the ghostscript interpreter (*gs*). The telepointer module (*tp*) provides a distributed, shared pointing device. The *mdb client* and *vcr client* allow for remote access to a multimedia database (MDB) and the VCRoD service, respectively. Two further modules are floor control (*fc*) and session control (*sc*), which both interface the collaborative services model (*csm*). Modules can be used, for instance, to extend dlb's media capabilities with new media types (html, jpeg, animations, etc.).

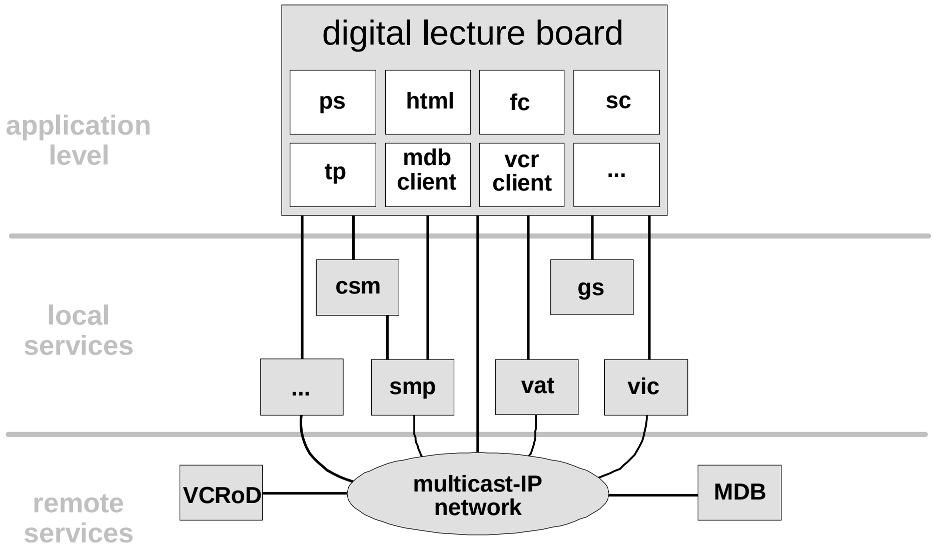

Fig. 3: Components of the dlb system architecture.

Besides embedded modules, the dlb makes use of *local services* such as ghostscript (*gs*) for rendering postscript pages, *vat* for audio communication, *vic* for video communication, the collaborative services model (*csm*) for managing collaborative services, and the scaleable multicast protocol (*smp*) for reliable multicast transmission. Remote services accessed via a network are, for instance, the Video Conference Recording on Demand service (*VCRoD*) or a multimedia database (*MDB*). Note that vic, vat, and gs are existing software components whereas csm and smp have been specifically developed in the context of the dlb. Instead of integrating these components into the dlb core, we implemented them as separated service applications which can be accessed by a well-defined application programming interface. We favored this approach for the following reasons: first, other applications can also access csm and smp services and, second, the implementation of the dlb is eased since these two service tasks have been separated from the dlb. On the other hand, separate services increase internal communication within the dlb system.

- **scaleable multicast protocol** (smp): At the beginning of our project, a commonly accepted reliable multicast protocol was not available on the Internet. Most protocols are either heavy-weight or they are integrated in applications [Floyd et al. 95]. So we decided to implement the scaleable multicast protocol smp which is based on the SRM paradigm (Scaleable Reliable Multicast) of wb but it is enhanced with a local scoping mechanism for better scalability. In contrast to SRM, smp is implemented in a separate process which provides reliable multicast transmission services to arbitrary applications. Smp offers different service classes, which satisfy the needs of different multicast group sizes, and it allows data sharing with an application in order to increase efficiency. Additional features are source ordering, late join, and rate control (e.g. for low bandwidth links). An application using smp does not have to be concerned with reliability issues. For more details on smp the reader is referred to [Grumann 97].
- **collaborative services model** (csm): As indicated in Chapter 3, collaborative services provide an electronic surrogate to compensate as far as possible for the lack of inter-personal communication channels. The csm implements enhanced floor control and session control mechanisms and policies. Floor control realizes concurrency control for interactive, synchronous cooperation between people by using the metaphor of a *floor*. A floor is basically a temporary permission to access and manipulate resources (e.g. a shared drawing area). Session control denotes the administration of multiple sessions with its participants and media. Session control increases social awareness in distributed work groups because members gain knowledge of each other and their status in the session. The csm keeps the collaboration state (the relationships between participants, floors, resources, and sessions) in a single object-oriented model. The model is replicated on each participant's workstation and held consistent by using an optimistic synchronization scheme.

Applications using csm get either messages about the collaboration state or they can explicitly send queries to csm, e.g. to ask if the floor for drawing on the shared workspace is currently available. Just as smp, csm is implemented in a separate process which provides services to arbitrary applications. This is specifically useful if several application are involved in the same session. The csm supports the following features:

- Administration of participants, groups, sub-groups, and super-groups,
- management of resources and assignment to participants, groups, and sessions,
- participants with different roles and privileges (e.g. teacher, student, etc.), and
- different floor control policies (e.g. implicit control, explicit control, chair control, etc.).

For a more detailed description of the csm component see [Hilt & Geyer 97].

6 Related Work

Besides various existing video conferencing systems such as NetMeeting, CUSeeMe, Intel's ProShare etc., which provide audio/video transmission, application sharing and standard whiteboard features, we know of two approaches closely related to ours. The „Authoring on the Fly" (AOF) [Bacher & Ottmann 96] concept merges broadcasting of lectures with authoring of CBT software. With AOF lectures are transmitted by means of an extended whiteboard to a number of receivers. Interactivity is limited to the audio and video channel, modifications to the transmitted material are not possible for receivers. Thus, collaborative types of instruction are not supported. The sender's (teacher's) media streams are recorded locally. The synchronized recording together with lecturer slides and additional media, such as animations, are then transformed to a CBT course which can be either published on a CD-ROM or accessed through the WWW. The Interactive Remote Instruction (IRI) system developed at Old Dominion University [Maly et al. 96] provides a very powerful, integrated teleteaching environment. The system can be used to view or make multimedia class presentations, to take notes in a multimedia notebook, and to interact via audio/video and shared tools. Furthermore, it provides class management and floor control. The system differs from ours in that IRI partly relies on analog transmission of NTSC video signals and that collaboration is limited to application sharing.

7 Conclusion

Experiences in the TeleTeaching project at the universities of Mannheim and Heidelberg indicate that standard video conferencing software is insufficient for the purpose of computer-based distance education. As a consequence, we decided to develop the digital lecture board (dlb), an integrated teleteaching tool, which takes into account the requirements of synchronous, remote instruction. The dlb supports multimedia presentations in RLR up to modern, collaborative types of instruction in RIS and IHL. As a part of a teaching and learning system, it allows for pre- and post-authoring of teaching and learning material. The current implementation presented in this paper already supports most of the envisaged features. In the upcoming semester, we intend to employ the dlb for a few trials in a distributed seminar for evaluation purposes. Future work will concentrate on the integration of audio and video communication in the dlb's user interface, and on extending dlb's media capabilities. Moreover, we also plan to work on integrating animation and simulation facilities.

8 References

[Bacher & Ottmann 96] Bacher, C., Ottmann, T. (1996): **Tools and Services for Authoring on the Fly**. In: Proceedings of ED-MEDIA'96, Boston.

[Eckert et al. 97] Eckert, A., Geyer, W. & Effelsberg, W. (1997): **A Distance Learning System for Higher Education Based on Telecommunications and Multimedia - A Compound Organizational, Pedagogical, and Technical Approach**. In: Proceedings of ED-MEDIA'97, Calgary, June 1997.

[Floyd et al. 95] Floyd, S., Jacobson, V., McCanne, S., Liu, C., Zhang, L. (1995): **A Reliable Multicast**

Framework for Light-weight Sessions and Application Level Framing. IEEE/ACM Transactions on Networking.

[Geyer et al.97] Geyer, W., Eckert, A. & Effelsberg, W. (1997): **Multimedia Technologie zur Unterstützung der Lehre an Hochschulen** (in German). To appear in: Multimediales Lernen in der beruflichen Bildung, Verlag BW, Nürnberg.

[Grumann 97] Grumann, M. (1997): **Entwurf und Implementierung eines zuverlässigen Multicast-Protokolls zur Unterstützung sicherer Gruppenkommunikation in einer TeleTeaching-Umgebung**. Master's Thesis (in German), Lehrstuhl Praktische Informatik IV, University of Mannheim.

[Hilt & Geyer 97] Hilt, V. & Geyer, W. (1997): **A Model for Collaborative Services in Distributed Learning Environments**. In: Proceedings of IDMS'97, Darmstadt, LNCS 1309, 364 -375.

[Holfelder 97] Holfelder, W. (1997): **Interactive Remote Recording and Playback of Multicast Videoconferences**. In: Proceedings of IDMS'97, Darmstadt, LNCS 1309, 450-463.

[Macedonia & Brutzman 94] Macedonia, M.R. & Brutzmann, D.P. (1994): **MBone Provides Audio and Video Across the Internet**. IEEE Computer. 27(4).

[Maly et al. 96] Maly, K., Wild, C., Overstreet, C., Abdel-Wahab, H., Gupta, A., Youssef, A., Stoica, E., Talla, R.,Prabhu, A. (1996): **Virtual Classrooms and Interactive Remote Instruction**. In: International Journal of Innovations in Education, 34(1), 44-51.

[Ousterhout 94] Ousterhout. J. K. (1994): **Tcl and Tk Toolkit**. Addison-Wesley.

[Parnes 97] Parnes, P. (1997): **mMOD: the multicast Media-on-Demand system**. Paper submitted to NOSSDAV'97, http://ctrl.cdt.luth.se/peppar/progs/mMOD/

[Schulzrinne et al. 96] Schulzrinne, H., Casner, S., Frederick, R., Jacobsen, V. (1996): **RTP: A Transport Protocol for Real-Time Applications**. Internet RfC 1889, IETF, Audio-Video Transport Working Group.

[Welch 97] Welch, B. B. (1997): **Practical Programming in Tcl and Tk**. 2nd edition, Prentice Hall, New Jersey.