An Interaction Control Architecture for Large Chairperson-controlled Conferences Over the Internet

Lukas Ruf, Thomas Walter, and Bernhard Plattner

Computer Engineering and Networks Laboratory
Swiss Federal Institute of Technology, ETH
ETH Zentrum Gloriastr. 35
8092 Zürich Switzerland
{ruf,walter,plattner}@tik.ee.ethz.ch

Abstract This paper presents a novel approach for interaction control in large, synchronous and loosely coupled but chairperson-controlled conferences. Based on the IP-Multicast protocol which is extended by mechanisms allowing resource reservation for prioritized flows, the proposed architecture supports interaction control among conference members as well as focus control of each conference participant.
Interaction control is performed by using a scalable signaling protocol: a conference may be recursively split into sub-sessions each of which provides an identical functionality independently of the others and establishes a singular session. A session is controlled by a chairperson who manually grants, revokes or rejects an interaction request of a registered session member. The granted interaction is announced to the session participants such that all participants may have their focus automatically set to the audio and video streams providing session participant.
The such provided focus control enables conference participants to individually manage their own audio and video stream perception, i.e. a participant either decides individually to whom the personal attention should be granted or follows strictly the session's focus granted by the chairperson.
The proposed conference control architecture provides the network-management platform to large virtual classrooms for university-like lectures over high-speed Internet-connections.
While focusing the application of the conference control architecture to university-like lectures, it remains generally applicable for any audio- and video-supported chairperson-controlled conference over the Internet.

1 Introduction

The use of information and communication technology in teaching and learning has initiated a transition to remotely sourced university lectures; [10,23,26] are a few examples. In such a scenario, lecturers and students are separated in space, i.e. students can attend lectures from almost everywhere and in particular from home. As already noticed by others it can be foreseen that this tendency will continuously increase [2,3,6]. The reasons are manifold: Professionals performing continuous life-long learning to keep in pace with the steady progress of research might be interested in reducing traveling time when attending a course. Students may prefer attending a lecture from home to extract the relevant information better. Educational institutions might be interested in reducing costs by transposing the ever demanding need for university space to virtual classrooms.

Teaching and learning in a virtual classroom can be seen as a large chairperson-controlled conference. It is a large conference since, in general, the number of participants is huge. It is chairperson-controlled because if a student intends to ask a question then she or he signals her or his intention to the lecturer. The lecturer grants or rejects attention to the asking student, and, thus, acts as a chairperson. In addition to asking questions, students should have the possibility of chatting with each other. Like in a traditional classroom setting this interaction should bypass the control of the lecturer. In order to create a feeling of being in a single virtual classroom we require also that interaction is supported by exchanging audio and video streams [7].

In any large conferences, a form of coordination is required in order to prevent significant information from disappearing in the vast “background noise” created by chatting conference members. A possibility to cope with this problem is to enable a single speaker to provide information while all other conference participants may either focus on the speaker or, alternatively, create smaller discussion-groups. For the coordination among sites we have designed an application-level protocol that supports interaction control among conference members as well as focus control of each conference participant. Similar to traditional lecture situations, a centralized management directs the mainstream of the lecture while enabling and disabling request-reply like questioning. This paper focuses on protocol requirements and specification of a conference control architecture to be used in a synchronous[1] distance education environment for university-lectures. Even more, the protocol can handle sub-conferences between groups of participants. For high-quality audio and video communication, the systems foresees the use of resource reservation. The proposed conference control protocol is generally applicable in any conference-like situation which requires audio and video support and is led by a chairperson.

1.1 Overview of this paper

In Section 2, we discuss assumptions and requirements of the assumed scenario. The protocol implementation is presented in Section 3. A discussion of related approaches is provided in Section 4. Section 5 provides an overview of the results and concludes the paper.

1.2 Terminology

In this paper we define our terminology as follows:

- Lecture participants are divided into *lecturer* and *students*. The term *lecturer* may be applied to any conference-controlling chairperson.
- The *local classroom* denotes the place where the lecturer is located.
- *Local students* attend a lecture in the local classroom.
- *Remote students* participate in the lecture over the Internet.
- The term *virtual class* denotes the aggregation of lecturer, local and remote students.
- An abstraction of the location of the virtual class is named *virtual classroom*.
- *Lecture-interaction* denotes a dialogue between students and the lecturer that is percept by the virtual class.
- The lecturer controls lecture-interactions by using a *lecture-controlling computer*.
- A *participating site* denotes a computer which is connected to the virtual classroom.

2 Basic Assumptions and Requirements

In this paper, we assume a distance education environment where students are in a local classroom as well as at remote sites. The virtual classroom is established over the Internet to which all remote students are connected. Each participating site runs a multimedia-capable computer which has attached camera, speakers and microphone. The computer is used for the following purposes: primarily, it receives audio and video data from the virtual classroom. Received audio and video are decoded, decompressed and presented to the corresponding end-devices. During interactions with the lecturer or other participants, the

[1] In synchronous conferences, participants are separated only in space. The asynchronous model further separates the participants in time.

student's computer acts as audio and video source. It compresses, encodes and sends audio and video data to the virtual classroom.

In the near future, available bandwidth to users at home will increase by using broadband cable TV or any mode of the digital subscriber line technology as the underlying link. Differentiated [4] and integrated service qualities are about to be installed and configured in edge-border routers of Internet service providers. High-quality audio and video distribution over the Internet will become reality and, thus, will allow students to actively participate in virtual classes from remote locations.

Observing traditional university-like[2] lectures, the lecturer is the acting person. The audience is listening while viewing teaching aids (e.g. slides) and looking at the lecturer. In a lecture, participants may require a temporary change of the lecture-attention (floor control) by requesting the focus of the lecture to ask a question. The other participants are aware of this interaction.

Separation in space should be bypassed in the virtual classroom so that the differences between a traditional and a virtual class are minimized. A major drawback of space-separated lecture participation is the inability to perform an individual discussion with a desk-neighbor. We therefore introduce the concept of an individual focus. Focus control allows a lecture participant to decide personally to whom it is granted: to the lecture or to the virtual desk-neighbor that wants to discuss an individual aspect (cf. Section 3.8).

Today, distance education platforms lack the possibility of controlling and setting the individual focus to the participants own point of interests. However, this should not be done on the cost of the information flow of the lecture but additionally, i.e. the personal focus can be set to a discussion or chat with other participants.

Summarizing, in a synchronous [3] interactive distance lecture environment the following requirements are to be met:

- A virtual classroom must be established to integrate fully the remote participants into the lecture.
- High-quality audio and video (a/v) data must be transmitted bi-directionally to overcome the space-separation.
- Floor control and focus control according to a lecture-like policy must be integrated.
- Interactivity with the lecturer and interactivity between remote participants must be supported.
- Scalability and flow prioritization are preferred over lossless transmission.
- Platform and application independence is required to keep track with the development and deployment of off-the-shelf products.
- A uniform management platform is needed that covers lecture participation and individual audio and video supported discussions among participants.

3 Implementation

Distance education systems as in use today, mainly cover the distribution of audio and video data from a lecturer to remote students [23]. Interaction between student and lecturer is performed by sending e-mails or asking lecture-assistance via textual chatting-features.

Some other distance education scenarios simply offer conference protocols [10] based on the MBONE [15] tools. Again others [5] try to integrate audio and video distribution and whiteboard-applications in a single, hand-tailored application and, therefore, miss

[2] By the term *university-like* we mainly describe the behavior of students: they decide on their own whether they want to attend a lecture, do not want to participate or temporarily leave the classroom for "playing cards". At universities, it is the freedom of thoughts and behavior that creates this extraordinary spirit.

[3] In contrast to tightly coupled ones (refer to [11]) where conference participation and information access is strictly controlled, the loosely coupled session model allows a broader and more opened joining semantics.

the required platform-independence to reach not only experienced users but beginners as well. Even extensions to applications and transmission of data require major applications changes.

Assuming huge numbers of participating students as in traditional lectures, the transition to lectures in the virtual classroom goes with the possibility of a major extension to the number of listening and actively participating students. Thus, a centralized approach to lecture membership- and floor-control like in [5] may impose a major hurdle to the aspect of scalability.

To summarize, currently available distance education platforms have the following drawbacks:

- No audio and video interactivity neither between remote students and the lecturer nor among remote students.
- No distance education-tailored platform supporting the conceptual behaviors of university-like lecturers in a virtual classroom.
- No application and platform independence.
- Restricted scalability.

Recognizing the drawbacks mentioned above, we propose an architecture for university-like lectures in a virtual classroom to manage *scalable session control*, *floor control* and thus *interaction control* based on the concept of an *individual focus*.

3.1 Conceptual Overview

Our control scheme follows strictly the procedures of students attending a university-lecture, i.e. the general focus of interest is set to the lecturer. A student that intends to ask a question signals the intention and waits until the lecturer grants the requested attention. Besides the ”official“ interaction with the lecturer, students may whisper with their virtual desk-neighbors without disturbing the main flow. Lecture-interactions as well as individual discussion-groups of remote students are performed in real-time using audio and video data-streams.

The provided distance education platform is implemented in a strictly layered-approach. The coordination and selection of high-bandwidth data streams are separated from input and output applications by providing a local gateway application, called focus-control tool (see Figure 1). Thereby, we can provide platform and application independence. Any application on any platform that provides the required interfaces with respect to networking infrastructure and data formats of the virtual classroom, can be installed on remote students' systems.

3.2 Abstractions

Our architecture is based on the following abstractions:

- *Session*: A session denotes a communication group.
- *Session Holder*: Every session is controlled by a session holder.
- *Sub-session*: Sessions may be ”forked“ into sub-sessions.
- *Parent Session*: The term parent session denotes the origin of a sub-session.
- *Floor*: The floor of a session denotes the transmitted information.
- *Floor Holder*: The floor holder provides the source of information within a session.
- *Participant*: A participant joined a session.
- *Registered Participant*: Only registered participants may actively participate, i.e. interact with the session.

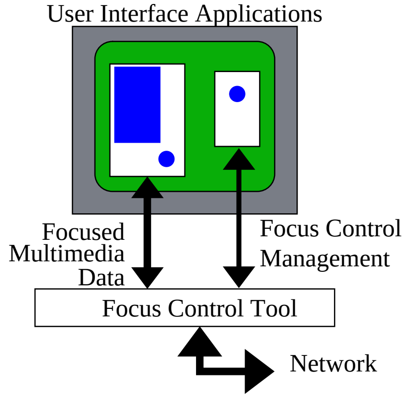

Figure 1. Focus Control Tool And User Interface

3.3 Communication Channels

The proposed conference control architecture manages the flow of data through four primary communication channels (see Figure 2):

- *SAC*: The *session announcement channel* provides the session description to join a session; see Section 3.5.
- *FDC*: By *the focused data channel*, audio and video data are received.
- *ICC*: By the *interaction control channel*, floor control is bilaterally coordinated but controlled by the chairperson of a session. While the session holder decides to whom the floor is granted, the decision is announced to future session holders in advance by the interaction control channel.
- *TAC*: The *teaching-aids channel* provides the flow for teaching aids (e.g. slides) and annotations to those.

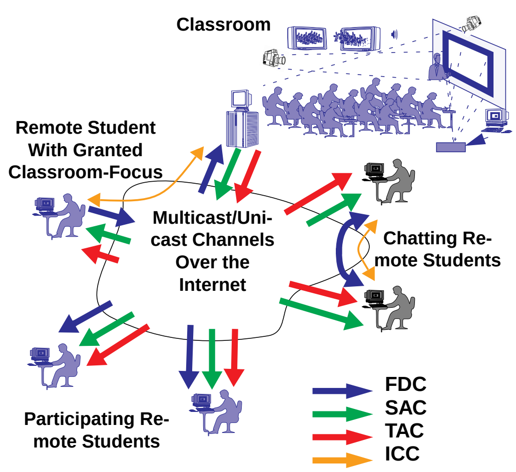

Figure 2. Channel Overview

Session Announcement Channel: If a newly created session's scope addresses a multicast [16] group, it becomes an open session. Registered participants of the parent session may join this sub-session. By the session announcement channel (SAC), participants may set their focus to the current floor holder whose audio and video is sent.

Focused Data Channel: Using the focused data channel (FDC), audio and video data are transmitted. An FDC exists per participating computer. It is dynamically set (cf. Section 3.7) according to the session announcements received by the SAC (cf. Section 3.3).

Interaction Control Channel: By the interaction control channel (ICC), the dialogue between participants is controlled (cf. Sections 3.8 and 3.4). The ICC is initiated by a participant that intends to interact with the session.

Teaching-aids Channel: The teaching-aids channel (TAC) exists only in the root session which is defined as the lecture itself (cf. Section 3.4). All participants of the virtual classroom continuously receive the teaching aids transmitted by this channel even if they participate in sub-sessions. The TAC remains valid during a teleteaching session.

3.4 Session Establishment

Conceptually, every session is established if a participant sends a *session request* to another participant (a state transition diagramme is shown in Figure 3). If the callee is willing to change his actual focus and to establish a sub-session, a *session grant* message is returned. Otherwise, the session request is *rejected*. In the established sub-session, the callee becomes the session holder, i.e. the chairperson. The callee and the session holder are marked as registered participants.

Both, the session request message as well as the session grant message, are manually initiated. The caller retrieves the session participants register (SPR) via a web-based interface and receives the address of the callee by selecting the appropriate entry. After having selected an entry, a session request message is automatically sent to the callee.

The callee now decides whether he wants to create a private, unicast session or an open, multicast one. In either case, the callee becomes the session holder.

If a multicast session is established, the session holder sends a registration message to the parent session. This message extends his or her entry in the parent SPR by the address of the newly created SAC. By representing this additional information, the callee becomes noticeable as a sub-session holder. The address of the appropriate SAC in the SPR is used to join a sub-session (cf. Section 3.5).

Neither registered nor un-registered session participants provide continuous video or audio streams to the floor. To deal with bandwidth constraints only floor holders are allowed to send their streams. By that, session participants are not continously aware of other attendees. Registered participants are marked within the SPR of the session holder.

A unicast session does not require a registration of the session holder in the parent's SPR; a unicast session in the virtual classroom represents the desk-neighbor whispering in a traditional lecture.

The sequence of messages sent during a session establishment is visualized in Figure 3. In both cases, the initially established channel to send a session request message to the session holder becomes the ICC during the interaction period (cf. Section 3.8).

At the beginning of the lecture, the lecturer and the lecture information itself are registered in the SPR of the web-page which belongs to the lecture.

3.5 Session Join

Joining a session requires the address of the appropriate SAC. This information is retrieved by traversing down the tree of sub-sessions starting at the root session's SPR or at the SPR

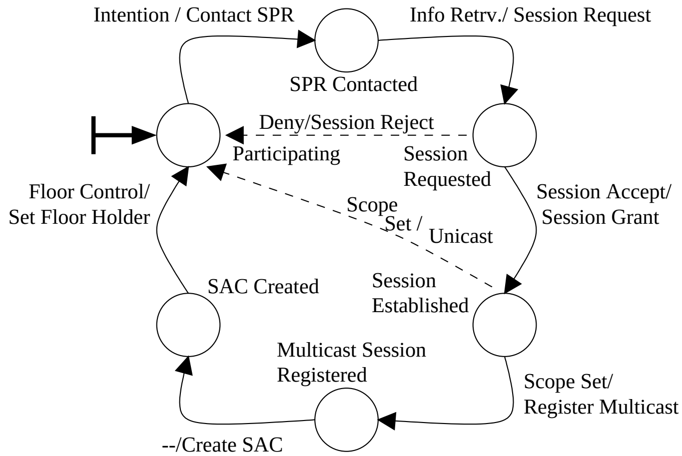

Figure 3. Message Sequence

of the current session if a sub-session should be joined. If an SPR is contacted for getting information of a sub-session, the requesting participant gets registered in the contacted SPR but not in the targeted sub-session, i.e. her or his participation is logged in the SPR. Clicking on the web-based SPR-representation transmits the information where to connect to the SAC of the targeted sub-session.

The information being transmitted on the SAC describe the sub-session as follows:

- Master session's SAC-address.
- Master session's TAC-address.
- Parent session's SAC-address.
- Floor group-address.
- Session holder.
- Floor holder.
- Validation checksum of the SAC-data.

The information to connect a SAC is used to perform a multicast-join operation into the SAC of the sub-session and, if applicable, into the TAC of the lecture. Receiving the information transmitted on the SAC provides the address of the floor, i.e. the FDC. The received data is used to set the focus of the participant to the current floor holder (see Section 3.7). Joining a sub-session changes the focus of the participant. However, teaching aids are always sent on the TAC.

3.6 Session Close

A session close is either performed individually by the session participant or, for all members of the session, by the session holder. The protocol provides the information to reconnect to the parent session or return to the root session.

If the session holder decides to close the session, this intend is signaled to the session before the SAC is closed. This session close-announcement is sent to allow the participants to have their focus automatically set to the parent session or to the root session otherwise.

If a participant intends to cancel its participation in a sub-session, he or she either returns to the parent or root session: a multicast-leave operation on the current FDC and SAC and reconnects to the appropriate SAC and FDC as described above. If a registration in the current session was once performed, the participation is de-registered if a controlled session close is performed by the session participant. Otherwise, if the participation is canceled by a system crash or abnormal program termination and the exiting participant was registered, the entry in the SPR is kept for the lifetime of the session. Session requests and join procedures (see Sections 3.4 and 3.5) to non-existent registrants are correctly handled by the focus control tool: no new session is established, the participation is kept unchanged.

3.7 Focusing

Focusing on the floor holder is performed by using the information provided on the SAC. The requirements for audio and video transmission impose quality of service constraints on bandwidth and delay. Thus, channel reservation for the receiver site is required as the individual focus-concept relys on information-channels. However, unwanted data may still arrive to the participant's computer. Therefore, the Focus Control Tool (see Section 3.8) provides a local filtering according to the currently set focus. The Resource Reservation Protocol (RSVP [18]) is used to setup quality of service constraints in edge-routers[4] at the participants' Internet service providers (ISPs). For scalability reasons, it is assumed that the backbones interconnecting the ISPs provide enough bandwidth and small delays to fulfill the requirements for live video and audio transmissions. Differentiated services [4] between ISPs will guarantee the required qualities for the virtual classroom. It is further assumed that backbones establish RSVP-tunnels for signaling purposes. As soon as a participant joins a session or recognizes an announcement-change, the current focus is teared down by sending the appropriate RSVP-TEARDOWN message to the routers; afterwards, an RSVP-RESERVE message is sent along the receiving path. RSVP-signaling is performed by the focus control tool (cf. section 3.8) independently of the user application outputting the received data. All this gives the required application independence in virtual classroom sessions.

3.8 Interaction Control and Floor control

Interaction control is performed by using the ICC. A participant that intends to acquire the floor establishes a unicast ICC to the session holder. A session participant becomes registered, if not already done, by sending a floor request. If a new session must be created, the channel established to send a session request to another participant becomes the ICC between the session participants (see Section 3.4).

By the ICC, a floor request is sent to the session holder. The session holder manually decides whether he or she wants to grant or refuse the floor to the requester. Refusing a floor request results in the termination of the ICC (see Section 3.6). If the floor is granted, a notification called *floor grant message*, is sent by the ICC to the requester. Besides the notification sent by the ICC, the change of the floor holder is, of course, announced on the SAC. The interaction policy imitates the known protocols of traditional university lectures: if the session holder intends to explicitly ask a participant, he or she asks the question by audio and video to the session and expects the callee to establish an ICC and follow the procedure as described above.

Until an interaction process is either closed by the session holder or canceled by the participant, the ICC remains established. Closing the ICC denotes the termination of the interaction. Using the ICC, request-reply alike dialogues are possible. It is further assumed that during discussion several ICCs per session are kept alive: the session holder decides "on the fly" to whom he or she wants to grant the floor.

If the session holder intends to ask a participant explicitly, the session holder articulates the question on the floor and addresses the participant by voice – as done nowadays in traditional university-like lectures – and expects that the participant responds by a floor request message as described above.

[4] By the term edge-router we denote the routers at the outer border of the Internet, i.e the ISPs, and not the interior routers of backbone networks.

The Focus-Control Tool (FCT) provides the gateway functionality as described above (see Figure 1). Session participation, floor and session request as granting both of them are handled by the focus control tool. It provides a clearly defined interface to allow graphical display tools being connected.

The FCT receives the SAC, FDC and TAC. After data streams passed filtering and validation tests, they are locally provided to the applications. This gateway functionality provides the above requested platform and application independence. If the display tools running under UNIX allow data reception for example via UNIX domain-sockets [9], filtered data are locally provided by this interface. If, on the other hand, display tools being started under other operating system only provide traditional IP-interfaces [8], valid data are transparently provided on such interfaces. Applications, originally not multicasting capable, may participate in the virtual classroom: the FCT transparently performs the unicast to multicast and vice versa gateway functionality. Intercepting the SAC-data, resource reservation signaling can be provided by the FCT to applications that are unaware of quality of service possibilities. The amount of transmitted audio and video data is reduced since FCT sends them out to the floor only if the floor is granted to that participant.

3.9 Teaching Aids and Annotations

The floor holder of the root session, the lecture, is allowed to provide annotations to the transmitted teaching aids. Since the floor focusing and filtering of the data streams is provided by the FCT, the teaching aids display- and modification tool could behave as if it is was a single-user application. Annotations to the teaching aids are added on the lecture-controlling computer. Modified information is sent to the participants of the lecture.

3.10 Media Scaling

By the term media scaling the transformation of input to output data regarding the available output-bandwidth is meant. Expecting the functionality and widespread use of active Internet nodes [1] to become real in the near future, we rely on the possibilities offered by node-plugins to perform the required media scaling so that a participating computer always receives the audio and video data in a best-possible quality.

3.11 Failure Handling

The proposed protocol follows the commonly available approach for reservation-failure handling in an Internet environment: If path-reservations fail, best-effort transmission is used per flow. Failure scenarios are identified for lost RSVP-reservation messages and the in-capability of routers on the path to provide the required resources. In case of a not established user-lecturer interaction, it is up to the student to signal the interaction-intent.

Currently, no service redundancy to provide failure recovery is foreseen in case of session controlling computer crashes. In a future generation of the currently provided virtual classroom description, failure recovery procedures must be integrated. A possible solution to provide the fault-tolerance might be the mirroring of the lecture-controlling computer. An in-depth analysis of this topic is subject of further studies.

4 Related Work

Besides the vast literature available on the topic of distance education, only few publications address the thematic of the underlying signaling protocol. Therefore, we concentrate on the ones that provided the most influence during the design of this protocol suite. The signaling protocols of the MBONE tools and the protocol concept of digital lecture boards (DLB), the ITU-T.120 recommendation as well as the MACS environment.

4.1 Session Description Protocol (SDP), Session Initiation Protocol (SIP), Session Announcement Protocol (SAP), Questionboard (QB)

The protocols SAP, SIP and SDP [19], [22], [20] are primarily designed to convey the information required to support the MBONE [15] tools. They are, spoken in general terms, the basics required to establish the session directory SD [21] and to be used by vic [25], vat [24] and wb [27]. By the use of the MBONE tools, loosely coupled conferences are created.

SAP provides the protocol to announce different sessions to be registered in the SD. It describes the information layout of the announced session. The principal purpose of SAP is the transmission of the SDP and SIP data. By SDP, the session is described in the SD. It provides the description of session participants to the SD which imposes limitations on scalability: explosion of messages. SIP, on the other hand, describes a method to invite computer users to participate in an MBONE session. Data transmitted provide information to start automatically the required applications if configured correctly.

The idea how to join a session in our architecture stems from [19]. The definition of the transmitted information on the SAC was influenced by SDP. We focused on the computer processing-performance aspects and therefore reduced the SAC-data to a binary and thus easier to process form while the MBONE signaling protocols are character-based. While SIP provides the invitation to join a session, we changed the direction and provide a more university-lecture oriented join mechanism. The concept of the SPR was influenced by the SD. We provide a web-based static form of the the information to join a session on the parent's session holder computer.

Floor control as provided in the herewith presented interaction control architecture is similar to the concepts evolved by the MBONE questionboard (qb) [14]. While qb provides moderated sessions where the floor is granted by the moderator as well, it limits its application due to the following reasons:

- Qb is designed for the MBONE. By that it relies on the conference bus [15] as designed for the MBONE tools. For proper operation, the MBONE tools are required.
- Qb does not allow the explicit revocation of the granted floor by the moderator.
- Hierarchical (sub-)session management is not provided.

Nevertheless, an aspect not covered by our approach resembles qb's recovery mechanisms by multicasting "hello-alike" host-startup messages.

The loosely coupled conference-join semantic of our conference control architecture is similar to the MBONE tools (IP multicast) while our centralized coordination resembles the original need for this protocol: the distance education environment.

4.2 The Upper Rhine Virtual University (VIROR)

In VIROR [26] the universities of Mannheim, Heidelberg, Freiburg and Karlsruhe in Germany developed a tele-teaching application-platform. They rely on the MBONE-tools to perform the protocol-oriented communication constraints and provide a tools-wrapping application. A drawback in their approach clearly is the dependency of the underlying operating system platform while not separating the management-flow from the user-interfacing information-flow.

In VIROR, evolved from the project Tele-Teaching Uni Mannheim-Heidelberg [5], floor control is implemented in a similar manner as our protocol does, but requires synchronized finite state machines on each participating computer. It further differs in the degree of interactivity between lecture participants; private chat-sessions between virtual desk-neighbors are not provided (see Section 3.8).

4.3 ITU Recommendation T.120 – Data Protocols for Multimedia Conferencing

The ITU-T.120 family of protocols [12] is designed to be used as basis for multimedia conferencing over public switched telephone networks. Similar to our approach, the ITU-T.120 family provides a centralized moderating functionality. In contrast to our interaction control architecture, the ITU T.120 family of protocols requires reliable communication. Multicasting is not intended. By that, this centralized approach provides explicitely, server-controlled connections per remote location. A specialized Multipoint Control Unit is required to provide the virtual interconnections among the single multimedia terminals. So, dynamic scalability is limited.

4.4 MACS – Modular Advanced Collaboration System

The approach taken by the project MACS [13] provides a similar concept as the one provided by our architecture. A floor and session control library performs the appropriate actions. This library provides an interface to allow application programmers to adapt their code. Portability is reached by the use of JAVA as library programming language. Currently, no hierarchical session management does exist.

While MACS provides a similar concept of establishing sessions, scalability limitations could exist in the tight management restrictions imposed by the registration mechanism. Hierarchical session management covered by a single user-interface is not foreseen. No integration of resource reservation protocols or differentiated service mechanisms is foreseen in contrast to our architecture (cf. section 3.7). Portability as provided by the JAVA-based library requires involved applications being adapted while our approach using the gateway functionality (cf. section 3.8) provides platform *and* application independence.

5 Conclusions

In this paper we have proposed a new virtual classroom architecture. The architecture provides interaction and floor control protocols to coordinate large audio and video conferences over high-speed Internet connections. Interactivity rules imitating university-lectures provide a scalable coordination platform for virtual classes. Scalability is reached by the distribution of coordination-competence to session holders, and the minimization of registration overhead. The approach of using native IP-multicast over the Internet supports a large number of receiving conference participants.

High-quality audio and video streams that enable participant to perceive everyone's interaction with the session holder are managed by using RSVP on the outer-border routers and differentiated services in the backbones. Audio-visually supported discussions between the lecturer and remote students as well as those among students establish a virtual classroom. Due to the loosely coupled organization of multimedia sessions and the proposed use of RSVP only on the edge border routers of the Internet, scalability of the provided architecture should be given. Further measurements of the deployed architecture within larger trials are required to prove the scalability.

Management is done by providing a control tool which acts as a gateway. Thus, a platform-independent control and management tool coordinates the reception of data flows from the virtual classroom as the session per se while off-the-shelf products can be used for multimedia input/output purposes. Altogether, our proposed coordination architecture and protocol cover the requirements put forward in Section 2 and deals with the problems mentioned in Section 3.

In this paper, aspects of security and membership control and effective memberships within a lecture (who is allowed to actively or passively participate) are not covered. The problem of multicast-address allocation is out of the scope of this paper. The reader is referred to [17] for further discussions.

The proposed platform will provide the network resource control to the project Easy Teach and Learn$^{(r)}$ [7] of the Communication Systems Group at Swiss Federal Institute of Technology.

6 Acknowledgments

We would like to express our acknowledgments to the Swiss Federal Institute of Technology (ETH Zurich) and the Hasler Stiftung for funding the project of Easy Teach and Learn.

References

1. Keller, R., et al.: “An Active Router Architecture for Multicast Video Distribution”, to appear in Infocom 2000
2. “Meeting Online Can Save Money, Boost Productivity”, ComputerWorld, 03/06/2000, http://www.computerworld.com/home/print.nsf/all/000306F3F2
3. “E-Learning: A Catalyst for Competition in Higher Education”, http://www.cisp.org/imp/june_99/06_99baer.html
4. Blake, S., et al.: An Architecture for Differentiated Services, IETF, RFC 2475
5. Geyer, W., et al.: The Digital Lecture Board – A Teaching and Learning Tool for Remote Instruction in Higher Education. ED-MEDIA/ED-TELECOM 99, Freiburg, BRD, 1998
6. ETH World, Swiss Federal Institute of Technology, http://www.ethworld.ethz.ch/
7. Walter, T., et al.:Easy Teach & Learn(r): A Web-based Adaptive Middleware for Creating Virtual Classrooms, to appear in HPCN2000
8. Stevens, W.R.,“Unix Network Programming, Network APIs: Sockets and XTI, Volume 1”, Prentice Hall, 1998
9. Stevens, W.R.,“Unix Network Programming, Interprocess Communications, Volume 2”, Prentice Hall, 1998
10. Interactive Remote Instruction, http://www.cs.odu.edu/ tele/iri/
11. ITU Recommendation H.245: Control Protocol for Multimedia Communication
12. ITU Recommendation T.120: Data Protocols for Multimedia Conferencing, 1996
13. Brand, O., et al.: MACS – Modular Advanced Collaboration System, Praxis der Informationsverarbeitung und Kommunikation, 22 (1999) 4, p. 213-220.
14. Malpani, R., Rowe, L.A.: Floor Control for Large-Scale MBone Seminars, ACM Multimedia’97, Seattle Washington, 1997
15. Macedonia, M. et al.: MBONE provides Audio and Video across the Internet, IEEE Computer, V27,4, April 1994
16. Deering, S.: Host Extensions for IP Multicasting, STD 5, RFC 1112, August 1989
17. Handley, M.: Session Directories and Scalable Internet Multicast Address Allocation. Proc.ACM Sigcomm 98, September 1998, Vancouver, Canada.
18. Zhang, L. et al.: RSVP: A new Resource Reservation Protocol, IEEE Network, Vol. 7, No. 5 (1993)
19. Handley, M., et al.: Session Announcement Protocol. IETF. Internet Draft draft-ietf-mmusic-sap-01.txt (1998)
20. Handley, M., et al.: SDP: Session Description Protocol. Network Working Group, RFC 2327 (1998)
21. Jacobson, V., Session Directory, Lawrence Berkeley Laboratory. Software on-line. ftp://ftp.ee.lbl.gov/conferencing/sd
22. Handley, M., et al.: SIP: Session Initiation Protocol. Network Working Group, RFC 2543 (1999)
23. Stanford Center for Professional Development, http://stanford-online.stanford.edu
24. Jacobson, V., et al.: vat, UNIX Manual Pages, Lawrence Berkeley Laboratory, Berkeley, CA.
25. McCanne, St., et al.: vic: A Flexible Framework for Packet Video, Lawrence Berkeley Laboratory, Berkeley, CA.
26. VIROR, The Upper Rhine Virtual University, http://www.viror.de
27. Jacobson, V., et al.: Using the LBL Network Whiteboard, Lawrence Berkeley Laboratory, Berkeley, CA.