

Comprehensive Multi-platform Collaboration

Kundan Singh, Xiaotao Wu, Jonathan Lennox and Henning Schulzrinne
Department of Computer Science, Columbia University, New York, USA

ABSTRACT

We describe the architecture and implementation of our comprehensive multi-platform collaboration framework known as Columbia InterNet Extensible Multimedia Architecture (CINEMA). It provides a distributed architecture for collaboration using synchronous communications like multimedia conferencing, instant messaging, shared web-browsing, and asynchronous communications like discussion forums, shared files, voice and video mails. It allows seamless integration with various communication means like telephones, IP phones, web and electronic mail. In addition, it provides value-added services such as call handling based on location information and presence status. The paper discusses the media services needed for collaborative environment, the components provided by CINEMA and the interaction among those components.

Keywords: collaborative work, multimedia communication, conferencing, SIP

1. INTRODUCTION

In many organizations, e-mail and tele-conferencing are the only means of collaboration. More recently, instant messaging (IM) is being used for short interactive communication. Even though these communication means are not designed for collaborative work, the limited set of available options causes them to put all their data such as meeting notes, documents, conference schedules and reminders into the email system.

We need a collaborative environment that seamlessly integrates with the existing communication means of email and phone as well as newer methods like IP telephony and instant messaging. Consider an IP telephony conference with some participants on phone, and some others using desktop audio/video clients. Late-arriving participants can browse through the past meeting proceedings, and non-participating group members can be automatically notified of meeting minutes and other important document locations via email.

One reason many earlier collaboration systems have not succeeded is that they were hard to use for people when the teams and groups span organizational boundaries. Also, they often require installing a lot of software, usually only available for limited set of platforms such as Windows, or work for only one vendor tools.¹ Collaboration tends to the “least common multiple” configuration supporting all needed tools and platforms, since groups can rarely say “sorry, since you cannot run this software, we will not include you in the committee”.

There are two modes of collaboration. A “synchronous” or tightly coupled collaboration is highly interactive and requires the active presence of the other members of the group. On the other hand, an “asynchronous” or loosely coupled collaboration is part of some collective activity directed towards some shared goal or common purpose, but does not require the active presence of the other members of the group. A comprehensive collaboration environment provides both synchronous and asynchronous collaboration tools and integrates the two so that users can easily alternate between the two.

Our system is different from other conferencing applications in that it integrates the two modes of collaboration. For example, same group of people can be addressed by video conference, IM and email, with appropriate archival of interactions. Secondly, it provides device-transparency by allowing access and interaction even if participants temporarily have only a phone or email. Although it is not new, we also provide hybrid interaction such that one can use phone for audio and PC for IM and document sharing in the same conference.

Our architecture provides building block tools for any type of multimedia collaboration, instead of focusing on specific types such as collaborative software development. We want to support three kinds of typical interactions:

Further author information: (Send correspondence to K.S.) K.S.: E-mail: kns10@cs.columbia.edu; X.W.: E-mail: xiaotaow@cs.columbia.edu; J.L.: E-mail: lennox@cs.columbia.edu; H.S.: E-mail: schulzrinne@cs.columbia.edu

long-lived distributed groups that alternate between synchronous and asynchronous interactions, such as design teams, college classes, committees and work teams, asymmetric events such as lecture and lecture series, where interaction is mostly limited to asking questions to the speaker, and short-lived spontaneous interaction among groups of people.

Our collaboration tools are based on standard protocols such as SIP² and Real-Time Streaming Protocol (RTSP³) for signaling, Real-time Transport Protocol (RTP⁴) for media transport, VoiceXML⁵ for voice-based interaction, Call Processing Language (CPL⁶) for network-based service creation, Language for End System Services (LESS⁷) for endpoint-based service creation and a web interface for asynchronous collaboration.

In this paper, we describe the architecture and implementation of our comprehensive multi-platform collaboration framework. While the previous work (⁸⁻¹⁰) focused on the telephony aspects, this article focuses on collaboration. We describe the requirements for comprehensive multimedia communication and collaboration environments in Section 2. Some related work is listed in Section 3. Section 4 provides an overview of the architecture and the user interface. Section 5 describes the synchronous collaboration architecture whereas Section 6 details the asynchronous collaboration. Finally, we present the conclusions and future work in Section 8.

2. REQUIREMENTS

The basic requirements for the comprehensive collaboration system consist of a personalized view of the system, real-time or interactive multimedia collaboration (called *synchronous*) and loosely tuned sharing of information (called *asynchronous*). A web-based user interface provides a portable and personalized way to access the system.

The per-user calendar for appointments and conferences should allow sharing, filtering and access control. The multi-party audio, video and text conferencing may also allow shared applications, access control, moderated conferences, recording and file sharing among the participants. Additional sharing of information via E-mail, voice or video mails should be possible. The various tools should be accessible from email or telephone, if possible.

3. RELATED WORK

The computer-supported collaborative work (CSCW) has been studied even before the web.¹¹⁻¹⁴ ACM's special interest group on supporting group work, SIGGROUP,¹⁵ explores topics related to computer-based systems that affects team or group in workplace settings. However, the focus remained mostly on web-based document sharing and concurrent editing in systems such as BSCW,¹⁶ Lotus Domino,¹⁷ Hyperwave¹⁸ or Livelink.¹⁹ Many researchers have explored specific types of collaboration such as collaborative software development,²⁰ electronic class rooms,²¹ network games and sharing health-care information.

Multimedia conferencing using audio, video, and data communication using IM and email, have independently evolved and become popular over the years.²²⁻²⁵ Using audio and video for collaborative work is not new.^{26,27} There are a number of audio/video collaboration systems such as MBone tools,^{28,29} MeetingPlace³⁰ and GnomeMeeting.³¹ The ITU-T's H.323^{32,33} provides video conferencing systems along with T.120 for data conferencing and T.128 for application sharing.³⁴

Most of the technologies used in our architecture, such as shared web-browsing,³⁵ conference floor control,³⁶ application sharing^{37,38} and web-based collaboration³⁹ have been investigated extensively. A number of web portals such as Yahoo! and MSN provide online calendaring, and sharing of information to some extent. However, the concept of group is rarely used. Our work is the first demonstration of a SIP-based comprehensive and extensible collaboration system. Our approach comes from a multimedia communication background, that extends the previous CINEMA communication suite¹⁰ to support different kinds of collaboration across different platforms. Our architecture integrates together the conferencing and collaborative computing approaches.

4. CINEMA ARCHITECTURE

The architecture consists of a set of distributed server components and user agents as shown in Fig. 1. The SIP registration and proxy server (sipd) is used for user location and forwarding of signaling messages. The multi-party conference server, sipconf,⁴⁰ forms the core of the synchronous collaboration infrastructure. The media server, rtspd, allows streaming of multimedia content for playback and recording. The unified messaging server, sipum, provides centralized answering machine, and multimedia mail service.⁴¹ A web-based interface provides asynchronous collaboration support. User agents such as regular PSTN phone via a SIP/PSTN gateway, IP-phone, or desktop based SIP user agents like sipc are used for synchronous collaboration. Interactive voice dialogue via the VoiceXML browser, sipvxml,⁴² allows easy access to a telephone user. The SIP server and the SQL database⁴³ form the core of the infrastructure for basic call flow (Fig. 2).

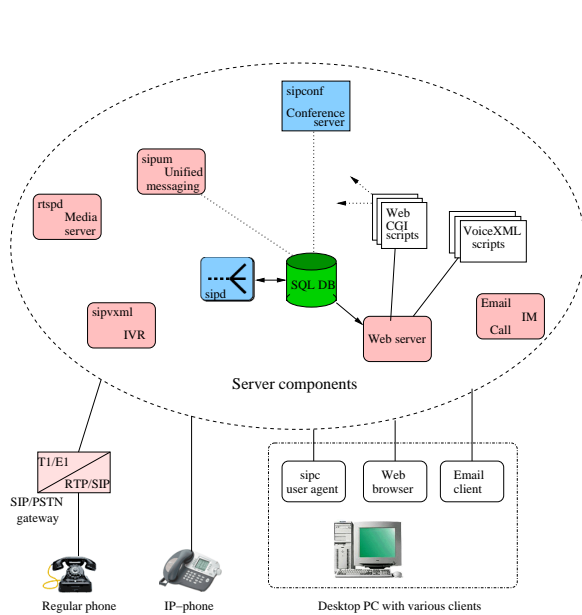


Figure 1. SIP-based collaborative work environment

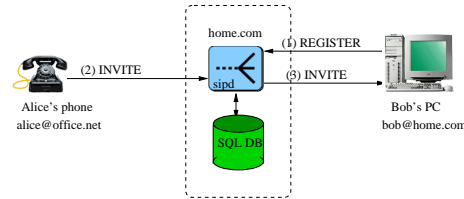


Figure 2. SIP call flow using proxy servers

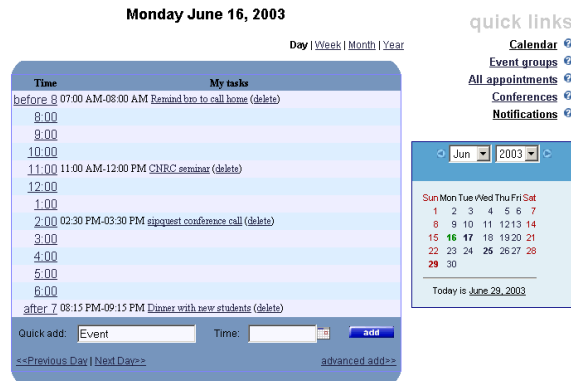


Figure 3. Personal calendar

4.1. Web interface

The web-based user interface allows managing user accounts, voice-mails and conferences. The web pages are generated using the HTTP CGI⁴⁴ scripts that access the SQL database for configuration and profiles. The web pages provide intuitive user interface components and context-sensitive help. There are multiple levels of details in different user expertise levels. For example, a beginner-level user accesses only basic features to get started whereas an advanced-level user can configure and manage detailed information. The interface allows configurable layout of the web pages so that a particular installation of the system can be adapted as per the service provider.

The call-routing profile is used to manage current phone locations, access control for incoming calls and programmable call handling. The unified messaging includes voice, video mails, emails and discussion forum. There is a per-user event calendar, address group and access group management. Finally the administrator interface allows configuring the servers, gateways, phone tariffs, and visual layout of the web pages.

4.2. Personal calendar and address book

When the user logs in from the web, it shows the most recent appointments and voice-mails. A personal calendar shows the various appointments or conferences scheduled for the user or his group (Fig. 3). The user can see the day, week, month or year view for different levels of information.

The per-user address book allows organizing the contacts into local or global access groups. A *local* group is visible only to the owner, e.g., “my friends”, whereas a *global* group is visible to everyone, e.g., “network research group”. An address book entry can belong to zero or more access groups. An event, such as an appointment or a class schedule, can have a group-name with given group-privileges. The read or write access privilege for an event can be **owner**, **group** or **everyone**, similar to Unix file permissions. The **read** access specifies who can view the description and details of the event. The **write** access tells who can modify the event attributes. A personal appointment typically has **owner** privileges for read and write, whereas a seminar series has **group** read access and **owner** write access.

4.3. Events and event-groups

An **event** is an individual event or appointment. An **eventgroup** is a collection of related events, e.g., an university course for which individual classes, or events, happen weekly. Every **event** can belong to an **eventgroup**. An **eventgroup** can have zero or more **events**. An **eventgroup** can optionally have a **repeat** indicator, e.g., every month, every year. The repeat indicator is useful if one does not want to itemize individual events, e.g., yearly birthday reminders.

An event-group may be associated with an optional conference name, e.g., on-line lecture series. While an **eventgroup** defines a group of events used in calendar, a conference is strictly a synchronous collaboration with additional attributes like supported media-types, dial-in number, recording formats, default audio sampling rate, public or private conference type and public or private participant list. Various SQL tables for storing the information are explained in CINEMA technical report.¹⁰

5. SYNCHRONOUS COLLABORATION

A multi-party multimedia conference is the simplest form of synchronous collaboration. In the absence of multicast, centralized conference servers provide an attractive solution for small to medium scale conferences.⁴⁰ Moreover, a centralized control integrates easily with other collaboration requirements such as floor control. For example, the organizer can control who gets to speak at any instant if there are multiple speakers, and enforce the policy at the server.

The participants dial the conference URL, e.g., *sip:staff-meet@cs.columbia.edu*, to join the dial-in conference. The conferences can be pre-scheduled from the web interface, or created on the fly, e.g., by dialing *sip:letsmeet.adhoc@conference-server*. For centralized conferencing, we need a central conference server such as sipconf and user agents such as sipc as described below.

5.1. User agent

Sipc is a SIP user agent that can be used for Internet telephony calls, multimedia conferences, presence, instant messaging, and shared web browsing. It supports a range of media types, such as audio, video, text and white board (Fig. 4), and can be easily extended to handle additional media types. It uses external media tools such as vic,⁴⁵ RAT⁴⁶ and wb.⁴⁷ We are currently developing our own low-latency audio tool. Beyond multimedia communication, it can also perform network appliance control, or act as SAP-based Internet radio or TV.⁴⁸ We are extending it to support emergency services.⁴⁹

The participants can also use other SIP-phones or regular telephones to join the conference. We are implementing another SIP user agent, sipz, for handheld devices to allow mobile multimedia participants.

5.2. Audio mixing

When the participants join the conference, the server mixes and redistributes the audio such that a participant hears everyone else except herself from the server. The server decodes the incoming audio from the participant, and puts it in a per-participant queue as shown in Fig. 5. On periodic interrupt, the participant audio is mixed, and redistributed back to the participant after encoding. Optimizations reduce the number of encoders and decoders.⁴⁰ The server acts as an RTP mixer⁴ for the audio. Each call leg in the conference forms an RTP session with the participant.

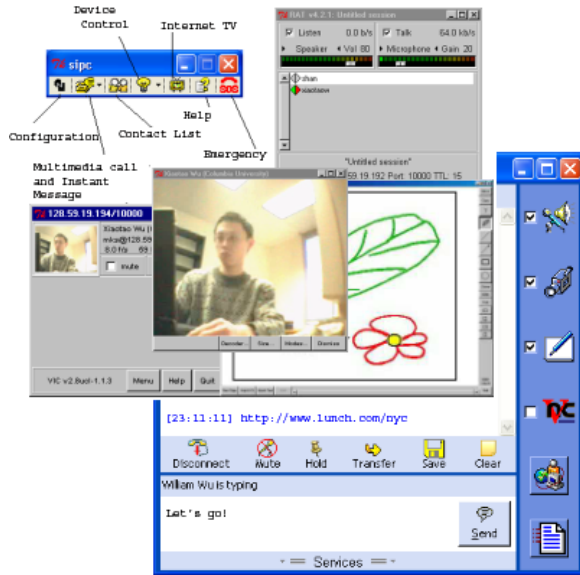


Figure 4. Columbia SIP User Agent (sipc)

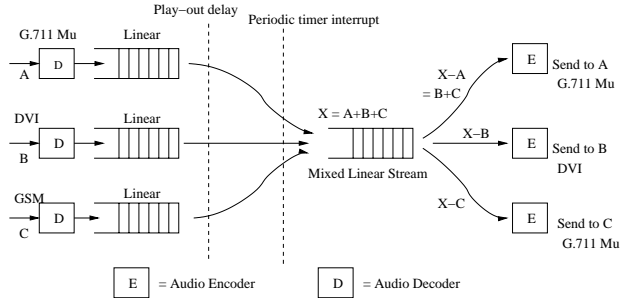


Figure 5. Audio mixing

```

MESSAGE sip:alice@office.net SIP/2.0
From: <sip:staff-meet@servers.com>
To: <sip:alice@office.net>; tag=Uo18a
Content-Type: Message/CPIM
...
From: Bob Wilson <im:bob@home.com>
To: Alice <im:alice@office.net>
Content-Type: text/plain
...
Meet me at Tom's at 8:00.

```

Figure 6. Example SIP MESSAGE for instant messaging

5.3. Video forwarding

Unlike audio, mixing does not make sense for video. Every participant may want video from everyone else in the conference. The server implements transparent packet forwarding for video. A video packet from a participant is distributed to every other participant in the conference without modification. In this case, the server does not implement the RTP stack for video session. The lip synchronization between the audio and video sessions is done at the participant's user agent on receiving the two streams.

5.4. White-board sharing

The White-board is a conference application for shared drawing. It allows synchronous collaboration using graphic information. The ITU-T Recommendation T.126 defines a protocol to manage the conference-wide synchronization of multi-plane and multi-view graphical workspace. However, in our system, we use an existing simple white-board application developed at UCL⁴⁷ in sipc and are planning to support the sharing in sipconf. The sipconf server simply forwards the drawing commands to all the participants except the sender. By this way, it does not need to maintain a shared graphical workspace internally. To allow new joiners getting historical drawings, the server can cache the drawing commands for late-arriving participants.

5.5. Instant messaging

The instant message (IM) handling in the conference server is similar to video forwarding. When *alice@office.net* sends an IM to *bob@home.com*, the SIP server at *home.com* domain proxies it to the current location of Bob's phone. An IM sent to the conference URL *sip:staff-meet@servers.com* is intended for all the conference participants. If the conference is not active or there is no other participant, then the server indicates the error to the sender. If the sender is not already in the conference, then the server can either indicate an error to the sender, or still continue to distribute the IM to the participants. In a way, the server provides a group address to send IM to, similar to email-groups. An example SIP MESSAGE sent by the server is shown in Fig. 6. The server can also forward indications⁵⁰ that allows Alice's user agent to display status such as "Bob is typing a message".

The server should allow transitioning from an IM session to a full multimedia session, and vice-versa, when the participant changes her media capabilities accordingly.

5.6. Shared web browsing

The SIP MESSAGE method can be used not only for instant messaging, but also for some additional control. For example, sipc can capture the browser event on navigation and indicate that HTTP URL to the remote party. The server forwards the message like any other IM, thus, readily supports sharing among multiple participants. In sipc, we have implemented an application to control Internet Explorer and Netscape for shared browsing.

5.7. Screen sharing

We have added support for the open source Virtual Network Computing (VNC⁵¹)-based screen sharing in both sipc and sipconf. VNC is a client server protocol, where the server shares its screen to a viewer or client. To avoid authenticating the client, we initiate the session from the VNC server to the listening client. If a participant shares her screen, her user agent invokes the VNC server application whereas all the other participants invoke the VNC client application. The conference server merely forwards packets similar to video forwarding. The data packets containing the screen buffers are forwarded from the VNC server to all the VNC client applications where as the control packets such as mouse and keyboard input are sent from the VNC client to the VNC server application.

5.8. Conference control

In a hybrid conference using phone for audio and PC for IM, it should be possible to control the conference from either phone or IM. Simple IM to the server can be used as control commands, e.g., if a participant sends IM text as “list”, the server returns the IM text containing list of all the active participants. Similarly, when a new participant joins or one leaves, all the existing participants are notified by the server via IM.

Conference floor control means controlling who gets the exclusive access of the shared media channels or resources. For example, typically only one participant should speak in a conference. In case of multiple contenders, the conference chair can decide who gets to speak. There are many ways to do advanced floor control such as using Simple Object Access Protocol (SOAP) to run Remote Procedure Calls (RPC)⁵² on the server, web interface, and via touch-tone phones. We are implementing the SOAP-based floor control in our server and user agent.

5.9. Dial-in vs dial-out conferences

Although most of our earlier discussion focused on dial-in conferences, dial-out mode is equally important. For example, if a participant wants to invite another user in the conference, or the server wants to send out call invitations to the intended participants at the scheduled time. Usually some form of audio and text announcement indicates the purpose of the call to the user. To avoid the dialed-out call going to answering machine, the server may prompt the user to press certain digits to actually join the conference.

6. ASYNCHRONOUS COLLABORATION

There are a number of related events during or after the conference, that need to be shared with others even when the conference is not active. For example, the recorded conversation or meeting minutes may be needed in subsequent meetings, off-line discussion on the topics covered in the conference needs to be co-ordinated in the same way as the conference was controlled or the notes may be edited remotely using WebDAV.⁵³ The primary objectives of these collaboration mechanisms are to avoid duplicating shared data and to provide some form of change control on shared data.

Use of RTSP, instead of the traditional download-and-view web model, enables the recording of the content once and the use of the pointer or the URL when forwarding the content without actually forwarding the multimedia file. This is desirable for low bandwidth situations where downloading a whole media file is very expensive, particularly if the recipient decides that she doesn't want to listen to the audio after hearing the first few seconds. Moreover, the multimedia content can be accessed with any RTSP based media player, e.g., Apple's QuickTime.

As mentioned earlier, every conference is associated with some eventgroup. An eventgroup can be associated with various forms of asynchronous collaboration mechanisms, such as file sharing and discussion forum. Conference participants can share meeting notes, agenda or other documents via the web.

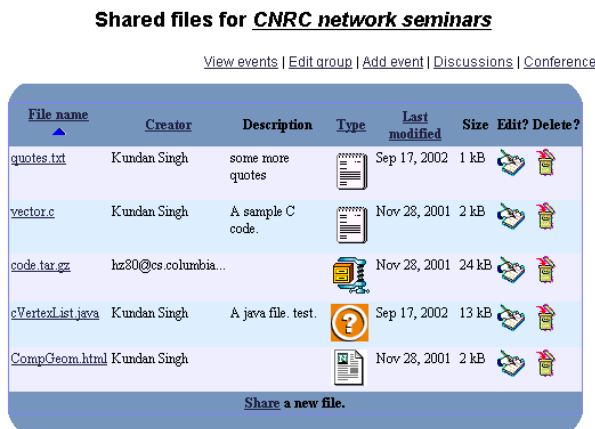


Figure 7. File sharing

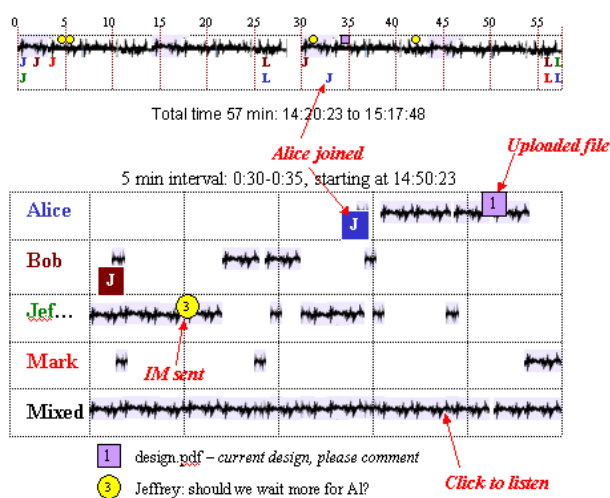


Figure 8. Web interface for conference recording

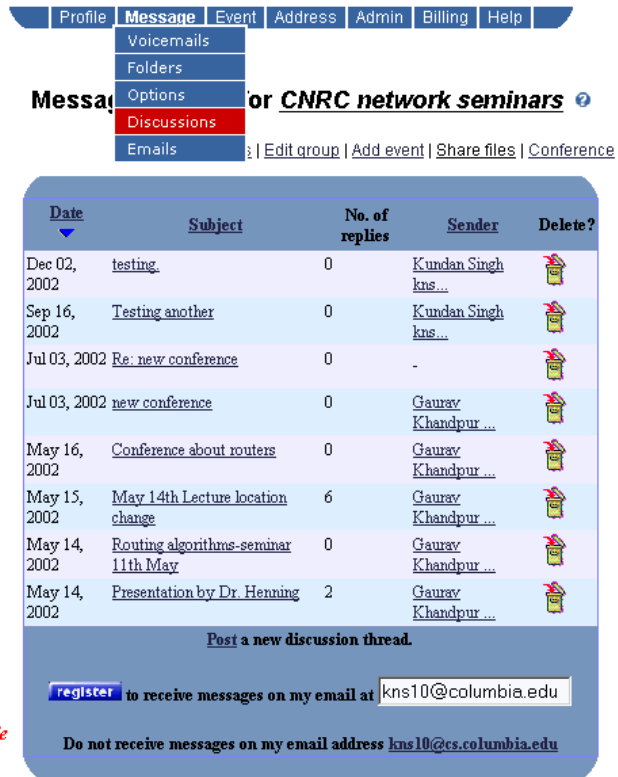


Figure 9. Web-based discussion forum

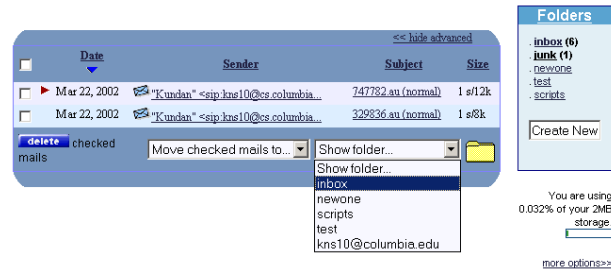


Figure 10. Voice messages

6.1. File sharing

The web interface allows uploading shared files as shown in Fig. 7. The users can register to get notified via email when the shared file is modified or deleted.

6.2. Discussion forum

Message boards and discussion forums facilitate asynchronous discussion on a particular topic. One advantage over email-based discussion is that it can systematically display the various discussion threads, postings and replies.

The users can also register to receive new posts or replies in their email. They can use email to post a message or reply to the discussion thread. Fig. 9 shows an example web interface.

6.3. Conference event recording

The system allows recording of the audio, video and IM communications in a conference. The audio recording at the conference can be done either when the media packets (RTP) are received from the participant or when the mixed stream is created as in Fig. 5. In the former case, the recording is done by dumping the raw RTP (and RTCP) packets along with packet size and time-stamp, in a file. On the other hand, a mixed audio stream can be recorded in standard Sun “snd” or Microsoft “wav” file format. Only rtpdump recording format is needed for video, since the server does not generate any mixed video stream. The system allows recording in a local file or to remote media server using an RTSP URL.

We are developing a web interface to display the conference proceedings in a time-line as shown in Fig. 8. The first time-line indicates the complete conference duration with the important events, such as the new user join, leave, file uploads and instant message interaction. The second time-line is the zoom-in view of a part of the conference duration as selected in the first time-line.

6.4. Unified messaging

The ability to send multimedia messages to other individuals or a group is an important feature of collaboration systems. Registered users can listen to their voice/video messages, recorded conference proceedings or view their emails from the web. An example web page is shown in Fig. 10.

The voice/video mail is recorded at the media server, *rtspd*, by the centralized answering machine and voice mail server, *sipum*.⁴¹ The server notifies the user of new incoming messages, e.g., using email, and indicates the pointer or URL to listen to the message. It should allow sending the media content instead of the pointer in the email, if the user wants that way.

6.5. Notifications and announcements

The system can notify the user of various appointment reminders, conferences schedules or changes in shared files, message board or incoming multimedia message. The user can schedule the same notification to multiple destinations.

It supports different kinds of notifications such as reminders for birthdays or appointments where the notification is generated before the event, wake-up call kind of one time notifications, and notifications for the eventgroup for its all individual events.

While an email or IM is an one-time event with no interaction, a phone-based notification can prompt the user with more options via IVR. For example, “press 1 to get notified again after 5 min, or press 2 to listen to the details of the event”. The system can allow scheduling the notifications from the web interface or via telephone using the touch-tone input.

7. ADDITIONAL SERVICES

There are other interesting services that assist both synchronous and asynchronous collaboration. For example, a conference server can dial-out a scheduled meeting only when all the required participants are on-line. An IM user can join a tele-conference and interact via speech-to-text and text-to-speech conversion between the IM text and other participants’ audio.

7.1. Presence

The presence information gets used quite often in people’s daily life. People are used to checking online status before starting a conversation with their IM “buddies”.

In our system, we base our presence information handling on the SIP event notification architecture.⁵⁴ Beyond the existing implementations’ presence status of *online*, *offline* or *away*, we consider both current and future communication availability via *timed-status*, a number of place-types, such as “home”, “office”, and “public” as well as a privacy classification into “public”, “private” and “quiet”.⁵⁵ The user can also indicate whether the communication is likely to be overheard or whether audio is considered undesirable.

The web interface displays the list of subscribed users (buddies) as well as all the others who are interested in knowing the presence status of this user as shown in Fig. 11. The client’s user interface is shown in Fig. 12.

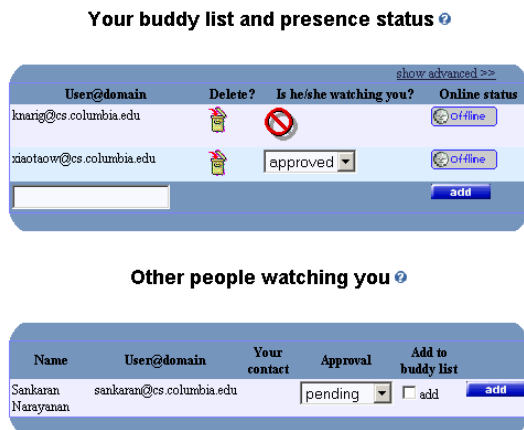


Figure 11. Web-based presence

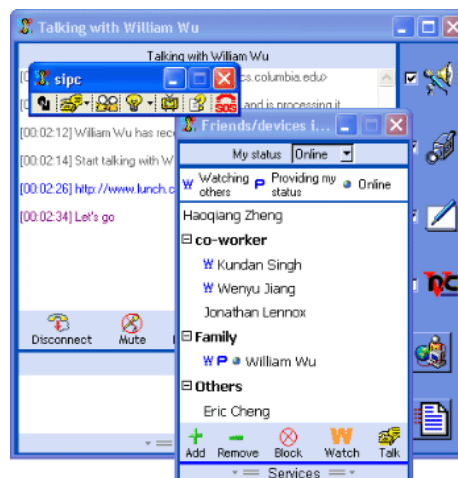


Figure 12. IM and Presence support in sipc

7.2. Interactive voice response (IVR)

We have discussed a number of examples involving user interaction via touch-tone input from a telephone. VoiceXML⁵ is an XML-based language developed by the W3C to create voice dialogs that feature synthesized speech, digitized audio, recognition of spoken and DTMF key input and recording of audio for telephony applications. Our sipvxml is a SIP-based VoiceXML browser that allows a SIP-phone, or a regular telephone via a gateway, to interact with the back-end application logic.⁴²

We have developed some CGI-based applications for voice-mail access and conference participation. Each registered user gets a unique telephone PIN (personal identification number) for authentication. The voice-mail script announces the number of new and old messages, and prompts the caller to listen to the messages.

7.3. Interaction among email, telephone and IM

Today, email is the most common form of electronic communication. However, the convenience of email is limited by the necessity of an Internet connected computer. A system that allows interworking of email with other communication means such as telephone or IM, will enhance user experience. Such system can be used to reach those users who only have email access via IM, define certain incoming emails as important and forward them to IM, get a virtual-IM account to interact with other IM users via email, access emails via phone, get notified of any important email on phone, and text-chat with other IM users or in a conference via phone.

8. CONCLUSIONS AND FUTURE WORK

We have discussed seamless integration between two types of collaboration modes: synchronous and asynchronous. The conference server and user agent in our CINEMA infrastructure allow synchronous multi-party multimedia collaboration via audio, video, instant message, screen sharing and shared web-browsing. The personalized user profile, calendaring, address book management, event and conference management, and system configuration can be done from the web interface. It also facilitates document sharing and asynchronous discussions among the group members. Moderators can monitor and control various synchronous and asynchronous activities. The messaging and notifications are used to reach the users when they are off-line.

A SIP-based architecture allows easily extending the infrastructure with new features, e.g., presence-enabled calls and programmable call routing. Interactive voice response provide easy access to the system from a telephone, whereas various text-to-speech tools allow interaction via plain email. This facilitates access to the system transparent to the end user device. Hence, we claim CINEMA to be a comprehensive multi-platform collaboration architecture. Moreover, the system allows hybrid interaction, e.g., phone for audio, PC for IM and document sharing in the same conference.

Although, CINEMA's main focus is on real-time synchronous communication, we also correlate the two modes of collaboration for an enhanced end-user experience. CINEMA can be used within an organization as well as in portal mode by application service providers. In future, we plan to support replicated databases, distributed mixing and other fail-over features such as automatic fail-over of a conference to a back-up mixer.

We have not implemented everything described in the paper. In particular, we are working on recording of conference events such as join or leave, programmable end system services, conference floor-control, performance measurement and improvement of the multimedia conference server, load balancing and scaling techniques for the servers, and optimizations for distributed multi-site collaboration. This paper is a continuation of our earlier work.⁸⁻¹⁰

Details on a number of important aspects such as security, email/IM integration, and database schema are skipped in this paper for space constraints. Interested readers can refer to the detailed technical report.⁵⁶

ACKNOWLEDGMENTS

Wenyu Jiang and Sankaran Narayanan contributed in the core design and implementation of CINEMA. A number of other students have contributed to various components in the architecture. The work* is supported by grant from SIPquest Inc.

REFERENCES

1. J. Schwartz, "Collaboration: More hype than reality," *InternetWeek (online newsletter)*, Oct. 1999. <http://www.internetweek.com/trans/tr99-bp1.htm>.
2. J. Rosenberg, H. Schulzrinne, G. Camarillo, A. R. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: session initiation protocol," RFC 3261, Internet Engineering Task Force, June 2002.
3. H. Schulzrinne, A. Rao, and R. Lanphier, "Real time streaming protocol (RTSP)," RFC 2326, Internet Engineering Task Force, Apr. 1998.
4. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," RFC 1889, Internet Engineering Task Force, Jan. 1996.
5. S. McGlashan, D. Burnett, J. Carter, S. Tryphonas, J. Ferrans, A. Hunt, B. Lucas, and B. Porter, "Voice extensible markup language (voicexml) version 2.0," tech. rep., World Wide Web Consortium (W3C), Feb. 2003. <http://www.w3.org/TR/voicexml20/>.
6. J. Lennox, X. Wu, and H. Schulzrinne, "CPL: a language for user control of Internet telephony services," internet draft, Internet Engineering Task Force, Aug. 2003. Work in progress.
7. X. Wu and H. Schulzrinne, "Programmable end system services using SIP," in *Conference Record of the International Conference on Communications (ICC)*, May 2003.
8. W. Jiang, J. Lennox, S. Narayanan, H. Schulzrinne, K. Singh, and X. Wu, "Integrating Internet telephony services," *IEEE Internet Computing* **6**, pp. 64-72, May 2002.
9. W. Jiang, J. Lennox, H. Schulzrinne, and K. Singh, "Towards junking the PBX: deploying IP telephony," in *Proc. International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, (Port Jefferson, New York), June 2001.
10. K. Singh, W. Jiang, J. Lennox, S. Narayanan, and H. Schulzrinne, "CINEMA: columbia internet extensible multimedia architecture," technical report CUCS-011-02, Department of Computer Science, Columbia University, New York, New York, May 2002.
11. M. Stefik, G. Foster, D. G. Bobrow, K. Kahn, S. Lanning, and L. Suchman, "Beyond the chalkboard: computer support for collaboration and problem solving in meetings," *Communications ACM* **30**, pp. 32-47, Jan. 1987.
12. J. Conklin, "Hypertext: An introduction and survey," in *Groupware — software for computer-supported cooperative work*, D. Marca and G. Bock, eds., IEEE Computer Society Press, 1992. IEEE Computer, September 1987.

*More information about CINEMA is at <http://www.cs.columbia.edu/IRT/cinema>

13. A. Dix, "Computer-supported cooperative work - a framework," in *Design Issues in CSCW*, Eds. D. Rosenburg and C. Hutchison, Springer Verlag, 1994. <http://www.comp.lancs.ac.uk/computing/users/dixa/papers/cscwframework94/>.
14. A. Dix, "Challenges and perspectives for cooperative work on the web," in *An International workshop on CSCW and the Web*, ERCIM/W4G, (Sankt Augustin, Germany), Feb. 1996. <http://orgwis.gmd.de/projects/W4G/proceedings/challenges.html>.
15. Association for Computing Machinery (ACM), "ACM special interest group on supporting group work (SIGGROUP)," 1996. <http://www.acm.org/siggroup/>.
16. W. Appelt, "WWW based collaboration with the BSCW system," in *SOFSEM (SOftware SEMinar)*, pp. 66–78, Springer-Verlag in the Lecture Notes in Computer Science 1725, (Milovy, Czech Republic), Nov. 1999. <http://bscw.gmd.de/Papers/SOFSEM99/sofsem.pdf>.
17. "Lotus domino." <http://www.lotus.com>.
18. "Hyperwave." <http://www.hyperwave.com>.
19. "Opentext corporation." <http://www.opentext.com/livelink>.
20. G. Kaiser and S. M. Kaplan, "CSCW and software process. session summary in ninth international software process workshop: The role of humans in the process," in *Ninth International Software Process Workshop*, pp. 9–11, Oct. 1994.
21. M. Mühlhäuser, "Interdisciplinary development of an electronic class and conference room," *Journal of Universal Computer Science (J.UCS)* **2**, pp. 694–710, Oct. 1996.
22. E. Schooler, S. Casner, and J. B. Postel, "Multimedia conferencing: Has it come of age?," in *24th Hawaii International Conference on System Science*, **3**, pp. 707–716, IEEE, (Hawaii), Jan. 1991.
23. M. Handel and J. Herbsleb, "What is chat doing in the workplace," in *Proceedings of ACM Conference on computer supported cooperative work(CSCW)*, (New Orleans, Louisiana, USA), Nov. 2002.
24. P. V. Rangan and D. C. Swinehart, "Software architecture for integration of video services in the etherphone environment," *IEEE Journal on Selected Areas in Communications* **9**, pp. 1395–1404, Dec. 1991.
25. S. Sarin, "Computer-based real-time conferencing systems," *IEEE Computer* **7**, pp. 33–45, Oct. 1985.
26. H. Schulzrinne, "Conferencing and collaborative computing," in *Dagstuhl Seminar on Fundamentals and Perspectives of Multimedia Systems*, (Dagstuhl Castle, Germany), July 1994.
27. E. A. Isaacs and J. C. Tang, "What video can and can't do for collaboration: a case study," in *ACM Multimedia*, pp. 199–206, (Anaheim, California), Aug. 1993.
28. S. McCanne and V. Jacobson, "vic: A flexible framework for packet video," in *ACM Multimedia*, Nov. 1995.
29. V. Kumar, *MBone: Interactive Multimedia On The Internet*, Macmillan Publishing (Simon & Schuster), 1995.
30. "MeetingPlace." <http://www.meetingplace.net/>.
31. "GnomeMeeting." <http://www.gnomemeeting.org>.
32. J. Toga and J. Ott, "ITU-T standardization activities for interactive multimedia communications on packet-based networks: H.323 and related recommendations," *Computer Networks and ISDN Systems* **31**, pp. 205–223, Feb. 1999.
33. J. Ott, "Teleconferencing in the ITU-T," in *IETF*, (San Jose, California), Dec. 1994. Multiparty Multimedia Session Control WG (MMusic), Talk (c).
34. P. Balaouras, I. Stavrakakis, and L. Merakos, "Potential and limitations of a teleteaching environment based on H.323 audio-visual communication systems," *Computer Networks* **34**, pp. 945–958, Dec. 2000.
35. S. Greenberg and M. Roseman, "Groupweb: A web browser as real-time groupware," in *Conference on human factors in computing systems, companion, proceedings*, pp. 271–272, ACM SIGCHI'96, (Vancouver, Canada), Apr. 1996.
36. H.-P. Dommel and J. J. Garcia-Luna-Aceves, "Floor control for multimedia conferencing and collaboration," *Multimedia Systems* **5**(1), pp. 23–38, 1997.
37. "pcAnywhere by Symantec, Inc.." <http://www.symantec.com/pcanywhere>.
38. "GoToMyPC by Expert City, Inc.." <http://www.gotomypc.com/>.
39. "VirtualPlaces." <http://www.vplaces.com/vpnet/index.html>.

40. K. Singh, G. Nair, and H. Schulzrinne, "Centralized conferencing using SIP," in *Internet Telephony Workshop*, (New York), Apr. 2001.
41. K. Singh and H. Schulzrinne, "Unified messaging using SIP and RTSP," in *IP Telecom Services Workshop*, pp. 31–37, (Atlanta, Georgia), Sept. 2000.
42. K. Singh, A. Nambi, and H. Schulzrinne, "Integrating VoiceXML with SIP services.," in *Conference Record of the International Conference on Communications (ICC)*, May 2003.
43. MySQL AB Co., "MySQL home page," <http://www.mysql.com>.
44. D. Robinson and K. Coar, "The common gateway interface (CGI) version 1.1," Internet Draft draft-coar-cgi-v11-04.txt,ps,, Internet Engineering Task Force, Oct. 2003. Work in progress.
45. UCB/LBNL, "vic – video conferencing tool." <http://www-nrg.ee.lbl.gov/vic/>.
46. UCL Multimedia, "Robust audio tool (RAT)." <http://www-mice.cs.ucl.ac.uk/multimedia/software/rat/>.
47. "Wbd: Whiteboard from University College London." <http://www-mice.cs.ucl.ac.uk/multimedia/software/wbd/>.
48. M. Handley, C. E. Perkins, and E. Whelan, "Session announcement protocol," RFC 2974, Internet Engineering Task Force, Oct. 2000.
49. H. Schulzrinne and K. Arabshian, "Providing emergency services in Internet telephony," *IEEE Internet Computing* **6**, pp. 39–47, May 2002.
50. H. Schulzrinne, "is-composing indication for instant messaging using the session initiation protocol (SIP)," internet draft, Internet Engineering Task Force, Feb. 2003. Work in progress.
51. T. Richardson, Q. Stafford-Fraser, K. R. Wood, and A. Hopper, "Virtual network computing," *IEEE Internet Computing* **2**, pp. 33–38, January/February 1998.
52. X. Wu *et al.*, "Use of session initiation protocol (SIP) and simple object access protocol (SOAP) for conference floor control protocol (SOAP) for conference floor control," internet draft, Internet Engineering Task Force, Mar. 2003. Work in progress.
53. Y. Goland, E. Whitehead, A. Faizi, S. Carter, and D. Jensen, "HTTP extensions for distributed authoring – WEBDAV," RFC 2518, Internet Engineering Task Force, Feb. 1999.
54. A. B. Roach, "Session initiation protocol (sip)-specific event notification," RFC 3265, Internet Engineering Task Force, June 2002.
55. H. Schulzrinne, "RPIDS – rich presence information data format for presence based on the session initiation protocol (SIP)," internet draft, Internet Engineering Task Force, July 2003. Work in progress.
56. K. Singh, X. Wu, J. Lennox, and H. Schulzrinne, "Comprehensive multi-platform collaboration," technical report CUCS-027-03, Department of Computer Science, Columbia University, New York, New York, Oct. 2003.