Web Conferencing Systems for Ubi-Media Collaboration: Achievements and Challenges

Bin Yu[1], Yong Rui
Microsoft Research
One Microsoft Way, Redmond, WA 98052, U.S.A.
{t-binyu, yongrui}@microsoft.com

Abstract

Web conferencing systems (WCSs) have entered a golden age of rapid development, as evidenced by more than a dozen commercial products and an expanding consumer population in the past five years. While WCSs have revolutionized the way people communicate and collaborate, there is still a lot of room for improvement in both the technologies and the users experience. In this paper, we summarize the latest achievements in WCS design and implementation from both scientific research and technical engineering, identify open questions and key challenges that deserve more research attention, and discuss interesting directions to explore for better solutions. We envision this paper will stimulate more researchers to explore ubi-media collaboration via WCS.

Keywords: Web conferencing systems, ubi-media collaboration

1. Motivation

A Web Conferencing System (WCS) is a combination of hardware/software utilities that help users to exchange information and collaborate across computer networks in real-time. Hardware utilities include input/output peripherals (e.g. video cameras, microphones, displays) and communication networks. Software utilities include PowerPoint presentations, web co-browsing, whiteboard sharing, application/desktop sharing, file transfer, text chatting, instant messaging, audio/video communication, polling, and session recording/playback. Figure 1 is a screenshot of Centra [4] from its user manual with illustration of some of the typical software utilities.

WCS has a wide range of applications, such as distant group collaboration, online training/education and marketing presentation. The long-term goal of WCS is to assist distant users to achieve commensurate or even higher level of productivity and security as in face-to-face meetings. The last decade has witnessed a remarkable series of developments in the area of WCSs. There are several factors behind it – strong growth of broadband population, rapid business expansion of U.S. companies into national and global markets, ever-increasing concern about travel cost and security, and, finally yet importantly, continuing research efforts from both academia and industry in providing better user experiences in collaboration. Numerous commercial systems have emerged on the market, driving the performance up and cost down. Now the user population of WCS has reached a critical mass: according to Frost & Sullivan [9], the web conferencing market stood at $472.1 million in 2003, and is expected to reach $3.02 billion by 2010. Among all the conferencing tools, e.g., audio, video and web, web conferencing is increasing the fastest.

There have been several reviews on existing WCSs, such as Robin Good [20], Think Of It [27] and Wainhouse Research [19], that follow latest development of new WCS features and evaluate performance of consumer products. However, we do not find an up-to-date scientific survey later than 1999 [26] that focus on the key innovations and open challenges in scientific research and technology improvement that are fundamental to the long-term prosperity of WCSs. Based on our first-hand experience in studying existing systems and developing new WCS features, we try to fill the gap by this survey that a) summarizes the latest achievements in WCS design and implementation from both scientific research and engineering enhancements, b) identifies open questions and key challenges that deserve more research attention, and c) discusses interesting directions to explore.

2. Discussion on Key WCS Facilities

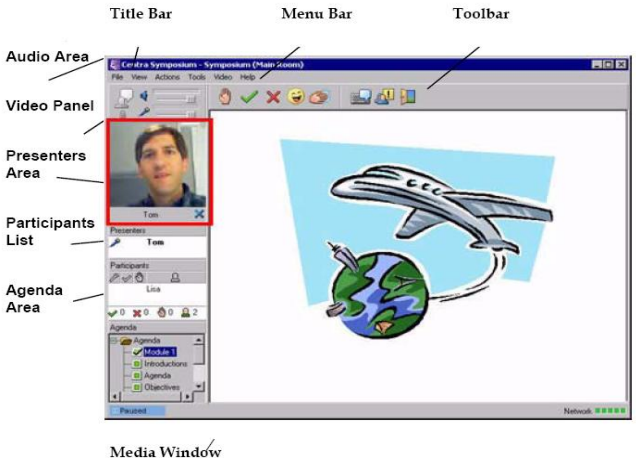

Figure 1. An example WCS interface from Centra [4].

[1] This project was done during the first author's internship at Microsoft Research. His current email is bin.yu.private@gmail.com.

2.1 Application/Desktop Sharing

Application/Desktop Sharing is a very powerful and extremely simple way of distributed collaboration. For example, Glance [10] is an outstanding WCS built solely on the desktop sharing facility. Although there have been some related research on advanced *collaboration-transparent systems* (see [7] for review), all commercial products adopt the centralized architecture: a single instance of the application runs at the initiating user's computer, local and remote users' input is assembled and feed into this instance, and the local graphic output is broadcast to remote hosts for display. This architecture is simple, and it avoids the overhead and copy right issues of installing software components on remote hosts.

Nevertheless, even with this simple setting, there are several challenging problems. The first problem is how to present clearer and smoother screen simulation at remote hosts. Existing systems send periodic updates of the change in graphic output of an application or the whole desktop, so the update rate is limited by available bandwidth and processing power. Essentially this can be modeled as the problem of how to efficiently stream a sequence of correlated images. Though applying complete video encoding/decoding codec suite may be too costly, some ideas behind video compression [14] can be borrowed. For example, if the origin of the screen output change is known (e.g. by examining the "repaint" message of the Windows messaging loop), advanced optimization is possible: when the change in screen output is caused by dragging a foreground window, the screen updates can be represented by motion vectors indicating the moving direction and distance of the foreground window instead of sending the actual pixels of the moving window.

Another challenge is floor control [12] among multiple users. For the case of one active user at any time, an interesting recent development is the activity sensing [15] protocol that borrows ideas from Ethernet-like conflict sensing techniques, though no user study is available yet to see how intuitive it is to average users. The case of concurrency control for multiple simultaneous active users is still an open challenge.

2.2 PowerPoint Presentation

PowerPoint Presentation facility is voted as the most important WCS feature in Web Seminarian newsletter's recent poll [28]. Existing systems normally use two methods for remote presentation. The first method is sharing the PowerPoint application. Besides the discussion on improving application sharing qualities in Subsection 2.1, more optimizations are possible by pre-analyzing the PowerPoint content. For example, screen output update can be more efficient by leveraging the knowledge on which part of the graphic output will change and in what manner.

The second implementation method is to *convert* the PowerPoint slides into other formats that can be presented in web browsers or similar viewers on various platforms, such as JPEG images, HTML pages, Flash or some proprietary format. Flash (as used by Convoq ASAP [6]) is very promising because it is an open standard and is capable of delivering high quality interactive rich media content with animations. One problem with such conversion methods is that it increases the loading time at startup and discourages impromptu idea sharing. To solve this problem, smarter conversion can be explored. For example, the conversion process can be *pipelined* so that the presentation can start quickly with the first slide or the first animation build while the following slides/builds are being converted. The second problem is that the converted format cannot support advanced features of PowerPoint, such as embedded objects (e.g. Microsoft Equation).

A promising direction to explore is to borrow the idea of *Polymorphism* from object oriented programming by playing the same PowerPoint content with the best presentation tool available on each host. For example, on a Windows computer with Microsoft PowerPoint installed, the presentation can be started as an ActiveX control in the Internet Explorer; on a Linux machine where OpenOffice is available, the presentation is embedded into Netscape as a Plug-In; on a computer with no PowerPoint viewing capability (which is a very rare case), the conversion or application sharing method can be utilized for that host alone.

2.3 Presence Awareness

Presence tools provide distributed users the rich presence information that is naturally given in face-to-face meetings. They are first introduced by instant messaging applications (e.g. MSN Messenger [17]), and now it has become an essential part of most WCSs. Current systems normally utilize a client side front-end that sends updates about local user's activities to a centralized directory server, and displays other users' information received from the server. The presence information is shown on a *buddy list* bar, including *availability information* (e.g. "available", "busy on the phone", or "do not disturb" as used by Convoq ASAP [6]), or more detailed *activity information* (e.g., "speaking" or "raising hand for question" as used by Meeting Central [18]).

There are two problems with such presence facilities. The first one is "not enough information", meaning that a user cannot get information about other users' activities with the same level of detail as in face-to-face meetings. Richer and more intuitive presence representation needs to be invented. For example, when two users are drawing on a whiteboard at the same time, a third user still cannot tell who is drawing what on the screen solely by looking at the

buddy list bar. However, if icons of the two users are shown beside each of their strokes, it will be much clearer.

The other problem is the opposite: "too much information". First, people do not want too much information to be revealed about their activities for privacy concerns. Second, people's attention space is limited and can only focus on certain amount of information before they get lost -- people do not want to be interrupted with too much uninteresting information. Such a *Dual Tradeoff* [23] problem has been an active field in the CSCW community: the tradeoff of *awareness vs. privacy* (how much information of the current user should be released to other users), and the tradeoff of *awareness vs. disturbance* (how much information of other users to present to the current user). The key challenge lies in how to develop a real-time filtering mechanism for presence information (both incoming and out-going) so that only interesting details are presented to users and uninteresting ones are hidden. This filter needs to be easy to customize, and it could automatically adapt by learning user preference.

2.4 Web Co-Browsing/Touring

Web co-browsing (or web touring) allows users to leverage existing information (e.g., company websites) and utilities (e.g., search engines and directories) on the web in their collaboration. Though it seems easy to support synchronous web browsing in a WCS, this turns out to be a big challenge that has not handled well by existing systems. Robin Good provides a list of "must-have" features that are not supported well [21]. The is due to both the intricacy of today's web page content (e.g. frame structures, scripts and server side processes) and complex client browser settings (e.g. Pop-up window filtering and whether scripting and active content are enabled). The solution may lies in a specially designed web browser that fully analyzes the requested web content so that exceptions are under control.

2.5 Audio Conferencing

With early WCS products, audio is provided by telephone-based conferencing solutions, which are reliable and provide great sound quality. A recent trend is Voice-Over-IP based audio conferencing, and some new WCS products support VoIP even with 14.4 kbps connections [29]. Because proprietary audio codecs are used, these systems generally support audio conferencing between users of the same VoIP provider and telephone users, while interoperability between different VoIP providers is still an open question.

2.6 Video Conferencing

After many years of research, experiments and discussion on video conferencing [3][8][1], people have come to a common understanding about video's role in group work – tele-data and tele-presence are both important. While t*ele-data* carries the information essential in getting the collaboration task done, *tele-presence* compensates for the gap between virtual and face-to-face meeting, such as managing the mechanics of conversations (e.g. turn taking), supporting non-verbal communication (e.g. gaze awareness and facial expressions), presenting real objects, etc. Despite the importance of tele-presence, video conferencing is still not an intensively used feature with today's WCSs. One major reason is that today's video conferencing technology cannot provide expected user experiences, and we will focus on several major technical issues below.

2.6.1 Networking Infrastructure and Signaling

Traditional video conferencing is based on the ITU-T H.320 [13] standard on top of circuit switched *ISDN* networks, which provide guaranteed bandwidth at 384Kbps. A recent alternative approach is provided by the ITU-T H.323 or SIP [24] standard based on *IP* networks, where no package delivery guarantee is supported but potential bandwidth upper bound can be much higher. For users who already have Internet connections and deem video as an add-on feature, the IP-based approach saves them from ISDN end-point installation and long-distance calling fee. However, ISDN video conferencing has been accepted and deployed internationally since 1990, and it provides guaranteed service with finely tuned quality. Therefore, both kinds of networks and related standards will co-exist in the near future.

2.6.2 Video Compression and Streaming

Many open video coding algorithms have been proposed [14]. The latest excitement has been the rapid development of MPEG4 AVC/H.264 standard, which supports very low bit rate video (even affordable by modem users with 56Kbps connections) at good quality, and pure software encoder/decoder libraries are available. Since many companies have independently implemented the H.264 standard, interoperability becomes an important issue [16].

2.6.3 Non-verbal Communication

Video can be used to convey facial expression, gestures and peripheral information about a remote user, but the most important element in face-to-face meetings is still very hard to support – eye contact. Half-silver mirrors are costly and bulky to use, while results from eye gaze correction [22] is not natural.

A fall-back solution is to support eye gaze awareness – letting users know who is looking at whom. There can be many ways to support gaze awareness, and we discuss two interesting solutions below. With the tele-immersion approach [11], view synthesis is applied to multiple camera streams to generate arbitrary perspective of each user. Then 3D images from all users are composed into a virtual scene similar to face-to-face meetings. This approach produces realistic conferencing experience, but the computation and streaming cost is relatively high. Another

solution is to use 3D avatars to replace images from real users to achieve similar effects with very low overhead (e.g. SmartMeeting [25]). This approach is very promising, and future graphics algorithms is expected to produce more realistic avatars that are generated based on real user images shoot from cameras.

2.7 Security

As more business meetings are shifted to be held in web conferencing rooms, several security issues become critical in WCS design. A good overview on web conferencing security practices is given in [5], and we focus on two key issues below.

2.7.1 Authentication

Email is widely used to represent user identity in existing systems. Normally the conferencing URL and a secret key are sent in an invitation email, and the invited users go to the URL and log in with their emails and the key. However, email can be vulnerable to many threats, such as eavesdropping, identity theft or even message modification, so messaging security mechanisms should be adopted, such as used in POST [2].

2.7.2 Data Protection

During a conferencing session, the real-time data communicated between distributed users should be protected against unauthorized access and modification. Encryption algorithms need to be selected carefully based on the specific type of data and application. For example, public key algorithms (e.g. RSA) are generally too slow to be employed for real-time interaction as compared to private key algorithms (e.g. DES). Video, audio and text content have different sensitivity, and video/audio streams are often strongly compressed, thus different encryption methods may apply (e.g. [30]).

In addition to encryption, other protection methods may be adopted, such as *VPN (Virtual Private Network)*, *SSL (Secure Sockets Layer)*, *Firewall* and NAT (*Network Address Translation)*. However, such measures may cause some conferencing systems to malfunction. For example, the T.120 and H.323 protocols dynamically assign port numbers, which increases the complexity of firewall configuration.

2.8 Session Recording and Playback

Currently only a few systems support recording and playback of conference sessions despite this facility's importance. We explain this as the lack of user interest caused by dissatisfactory user experiences. Current systems normally provide an option to "enable recording" when scheduling a meeting, and the meeting is recorded as image sequences of the screen output and stored on the recording server. There are several problems with such a video-based approach. The first problem is that the recorded video is typically of high bitrate, and it takes a long time to download it for reviewing for users with narrow connections. In addition, the conversion from a WCS meeting to a video recording is invertible and a lot of semantic information is lost. For example, when a user types a sentence as a comment over a shared application window, the video recorded does not keep the timestamp and the text of the user comment. Thirdly, the video recording content is hard to edit. For example, sales presentations normally have to be recorded repeatedly with minor modifications for perfection. If a few words have to be changed in a PowerPoint file, the whole presentation may have to be recorded again.

Our vision is that the user actions will be recorded as meta-data associated with the original data content in future systems. One possibility is to record tele-data sessions entirely by events. An *event* is the change of the tele-data as a conference session progresses. For example, for a PowerPoint presentation, events can be "next slide", "previous animation build", "laser pointer near the text box 'web' ", etc. Similarly, for a web touring session, events can be "go to URL http://...", "follow the link with text 'conferencing…'…" etc. If all the events of a conference session are recorded with timestamps, this session can be completely *replayed* by simulating the same sequence of events happening again onto the original data set accessed in the meeting. This way, a "recording" is the set of original data plus a text log of all the events. Session browsing, searching and summarization are all made much easier because the original user actions and associated data are recorded in textual format. In addition, minor changes can be made directly to the data files, and during playback the event sequence will be applied to the new content. Of course, there are several open issues associated with this kind of event-based recording/playback that need to be explored. For one example, some conferencing sessions cannot be expressed in discrete event sequences, such as sharing of an arbitrary application, so the challenge is how to combine event based recordings with video-based recordings seamlessly.

3. Conclusion

WCS is one of the key solutions to improving information workers' productivity in their daily work. This is especially important as more and more teams and projects are going global and distributed. While WCSs made significant progress during the past few years, there still exist great potentials for improvement. Based on our first-hand experience with existing WCSs and research on new WCS features, we summarized what existing WCSs had done well in the past and what could be done better in the future. We envision this paper will stimulate more focused research effort on Ubi-media collaboration by improving WCS design/implementation from both researchers and practitioners.

4. Reference

[1] A. Anderson, L. Smallwood, R. MacDonald J. Mullin and A. Fleming, “Video Data and Video Links in Mediated Communication: What Do Users Value?”, International Journal of Human-Computer Studies 52(1), pp 165-187, 2000.

[2] A. Mislove, A. Post, C. Reis, P. Willmann, P. Druschel and Dan S. Wallach, "POST: A Secure, Resilient, Cooperative Messaging System", HotOS 2003.

[3] C. Egido. “Video Conferencing as a Technology to Support Group Work: A Review of its Failures”, CSCW 1988.

[4] Centra, http://www.centra.com/

[5] Cisco, “Best Practices in Web Conferencing Security”, white paper.

[6] Convoq ASAP, http://www.convoq.com/

[7] D. Li and R. Li, “Transparent Sharing and Interoperation of Heterogeneous Single-user Applications”, CSCW 2002.

[8] Ellen A. Isaacs and John C. Tang, "What Video Can and Cannot Do for Collaboration: A Case Study", Multimedia Systems, Vol. 2, pp 63-73, 1994

[9] Frost and Sullivan, http://www.frost.com/

[10] Glance, http://www.glance.net/

[11] H. H. Baker, T. Malzbender, N. Bhatti, D. Tanguay, I. Sobel, D. Gelb, M. E. Goss, J. MacCormick, K. Yuasa and W. Bruce Culbertson, “Computation and Performance Issues in Coliseum: An Immersive Videoconferencing System”, ACM MM 2003.

[12] H.P. Dommel and J.J. Garcia-Luna-Aceves, “Efficacy of floor control protocols in distributed multimedia collaboration”, Cluster Computing, vol. 2, iss. 1, pp 17-33, 1999.

[13] “ITU-T”, http://www.itu.int/ITU-T/

[14] J.G. Apostolopoulos, W. Tan, and S.J. Wee, "Video Streaming: Concepts, Algorithms, and Systems", HPL-2002-260.

[15] K. Katrinis, G. Parissidis and B.Plattner, "Activity Sensing Floor Control in Multimedia Collaborative Applications", DMS 2004.

[16] MPEGIF Interoperability Test, http://www.mpegif.org/public/interop/index.php

[17] MSN Messenger, http://messenger.msn.com/

[18] N. Yankelovich, W. Walker, P. Roberts, M. Wessler, J. Kaplan and J. Provino, "Meeting Central: Making Distributed Meetings More Effective", CSCW2004.

[19] “Rich Media Conferencing 2004”, http://www.wainhouse.com/reports/rmc2004.html

[20] Robin Good, “Robin Good's official guide to web conferencing and live presentation tools”, http://www.masternewmedia.org/reports/webconferencing/guide/

[21] Robin Good, "Towards A New Generation Of Web Conferencing Tools: Co-browsing And Web Touring", http://www.kolabora.com/news/2004/11/29/towards_a_new_generation_of.htm

[22] R. Yang and Z. Zhang. "Eye Gaze Correction with Stereovision for Video Tele-Conferencing". ECCV 2002.

[23] S.E. Hudson and I. Smith, "Techniques for Addressing Fundamental Privacy and Disruption Tradeoffs in Awareness Support Systems", CSCW 1996.

[24] “SIP versus H.323”, http://www.iptel.org/info/trends/sip.html

[25] SmartMeeting, http://www.smartmeeting.com/

[26] S. Terzis and P. Nixon, "Building the Next Generation Groupware: A Survey of Groupware and Its Impact On the Virtual Enterprise", TCD-CS-1999.

[27] Think Of It, http://www.thinkofit.com/webconf/index.htm

[28] WebSeminarian, http://www.webseminarian.com/opinion/polls.html

[29] WebSeminarian, "Is Quality Voice-over-IP Ready For Prime Time?", http://www.webseminarian.com/opinion/VoiceOverIP.html

[30] Z. Liu, X. Li and Z. Dong, "Enhancing Security of Frequency Domain Video Encryption", ACM MM 2004.