Realising Real Time Multimedia Groupware on the Web

Colin Allison, Martin Bateman, Alan Ruddle

University of St Andrews
St Andrews, KY16 9SS
{ca,mb,alanr}@dcs.st-andrews.ac.uk

Abstract. Real-time audio and video conferencing has not yet been satisfactorily integrated into web-based groupware environments. Conferencing tools are at best only loosely linked to other parts of a shared working environment. This is mainly due to the special Quality of Service (QoS) requirements that these types of resources demand. This paper describes an approach to overcoming this problem by integrating the management of video and audio conferences into the resource allocation mechanism of an existing web-based groupware framework. The issue of adaptation is discussed and a means of initialising multimedia session parameters based on predicted QoS is described. In addition to linking audio and video media rendering quality to the predicted network QoS, the use of models to reduce bandwidth is also explored, replacing video content with control data. Component technologies utilised include the TAGS groupware framework, the Java Media Framework, a Conference Control Architecture and VRML-based avatars that can intermix with real actors.

1 Introduction

Web-oriented component technologies adhere to the 'anytime anywhere' requirements of distributed groupware and are largely platform neutral. The growth of the web as the preferred medium for group work has meant that, while the underlying principles of distributed groupware remain, there is now a clearly preferred means for its implementation and deployment. Systems which involve audio and video conferencing channels in addition to shared objects are referred to as real-time distributed groupware (RDG). Although a significant amount of groupware has been developed as embedded web applications, audio and video conferencing is still very much a bolt-on, and not at all satisfactorily integrated, or even possible to integrate. So, pre-web RDG exemplars such as the Warp shared spreadsheet [1] are particularly difficult to realise on the web[1].

[1] The Warp-based RDG exemplar was a multi-user, distributed, non-blocking shared spreadsheet object augmented with multicast audio and video channels. It used X-windows, IP Multicast, and RPC2, and is therefore an example of pre-web groupware. It was also a

The TAGS framework for groupware [2,3] differs from Warp in that it is completely web-oriented. It is currently used to support the development of groupware in the form of collaborative multi-user resources for teaching and learning. It has however proved difficult to incorporate real-time audio and video conferences into the system. Efforts to date have resorted to starting applications such as vic [4] and rat [5] either totally independently or from a browser script. This is not satisfactory as it means that (i) the session is not integrated in any useful way with the groupware resource allocation mechanisms, and (ii) a degree of technical expertise beyond that found in typical web users is required. This paper describes an approach to tackling this problem by using TAGS in conjunction with the Java Media Framework, a Conference Control Architecture and VRML-based Avatars that can intermix with real actors. The following sections describe the TAGS resource allocation mechanism, the Java Media Framework (JMF), QoS issues, adaptation strategies, a Conference Control Architecture and the role of Avatars in QoS-aware groupware environments.

2 Groupware Resource Allocation

TAGS, the Tutors and Groups Support scheme, is a project that researches, develops and deploys distributed learning environments. In TAGS groups form the basis of (i) privileges and access control, (ii) information dissemination and event awareness and (iii) allocation of shared resources. A home page, or portal, is generated dynamically for each TAGS user, and presents access via hyperlinks to all the resources allocated to all the groups that that an individual belongs to. TAGS provides three main abstractions for tutors: users, groups and resources. Each user is a member of one or more groups and each group is allocated zero or more resources. The concept of a resource is deliberately loose. It can be a simple timetable, an automated assessment exercise or an interactive multi-user simulation.

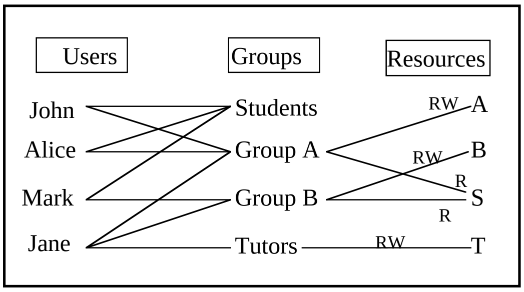

Fig. 1. Group Resource Allocation

In practical terms, tutors construct a collaborative learning environment by using the Users, Groups and Resources management tool. This sets up arbitrary relationships between users and resources, using groups as the basis for the mapping. Users and groups are unique by name; resources are unique by name and type. Access

system that required a significant degree of technical expertise to install and use, in strong contrast to the ease of use and deployment associated with web-based approaches.

rights can be specified when a resource is allocated to a group. A resource may simply be distinguished as Read-only or Read-Write, or it may export a more subtle set of access methods. This allows a particular resource instance to be allocated with different access methods to different users. Figure 1 illustrates an example set of relationships between users, groups and resources. John, Alice and Mark are members of the Students group. There are no resources shown allocated to the Students group. Jane is a Tutor and has created Groups A and B. John, Alice and Jane are in Group A and have resources A and S allocated. S is allocated to group A on a Read-only basis, whereas A is Read-Write. Mark and Jane are in Group B, which can access resource B as Read-Write and resource S as Read-only.

Could this allocation model be used to manage video conferences, as shared resources? The IP Real Time Protocol [6] (RTP) supports multicast *sessions* and participants, where a session is a short-lived entity which consists of members consisting of IP hosts specified by a mixture of multicast and unicast addresses and port numbers. In order to interoperate with the TAGS framework, there must be functionality to allocate the sessions to groups from within TAGS. This would mean that multimedia sessions are easy to setup and join, since the complexity of multicast addresses and port numbers is abstracted away from the user. The traditional method of joining multimedia conferencing session on the MBone [7] is via the Session Directory protocol, implemented by the `sdr` tool. SDR displays a list of sessions which are either scheduled to take place or which are currently underway. The entries consist of a multicast IP address, a port number, the multicast scope in the form of a Time To Live (TTL) value, the RTP payload type and some textual information about the session. From this information any session advertised by sdr may be joined. The mechanism we propose replaces the need to use `sdr` when joining a multicast session from within the TAGS framework, although it does not eliminate the need for the session directory protocol.

3 The Java Media Framework

The Java Media Framework (JMF) [8] is a collection of APIs which aim to provide a method for handling time based multimedia within Java. It does this by allowing the capture, transmission, storage and displaying of various formats of audio and video. There are two main distributions for the JMF, a pure Java implementation and a native library version. The pure Java implementation is limited in that it is unable to capture or transmit video/audio. It can however, render audio/video streams that it receives. In the native version the majority of the processing is performed by platform specific code. This has significant performance advantages over a pure Java implementation and offers greater capabilities for the programmer and user. A simple JMF application has three main parts. There is (i) a DataSource, which receives multimedia data from devices such as video capture cards or sounds cards; (ii) a Processor that can be used to change the format, frame rate, image depth or bitrate of the multimedia data - for example, video could be encoded to H.263 at 23 fps 120Kbps; (iii) a DataSink which is used when sending the data to its final destination - this could be to the screen, to a file, or the network.

JMF supports various network protocols for the transmission of multimedia. The most interesting of these from this projects point of view is multicast RTP[6] which can be used to disseminate realtime multimedia to a group of users. JMF's major advantage when compared to other video conferencing technologies is that since it is Java based, it can be incorporated into applets that can then be integrated into a web environment. There are several ways in which the bandwidth used within a conferencing session can be varied. These include, video frame rate, video size, image resolution, image depth, audio resolution, audio sample rate, changing codec and changing media. Table 1 shows the RTP payload types that are supported by the default installation of the JMF. Additional multiplexers, demultiplexers, filters and codecs can be added to the JMF via means of a plug-in architecture. Plug-ins must be registered on the host machine before they can be used.

A/V	Media Encoding	Cross Platform	Native Library	Bit Rate (sample)
A	G.711 (U-law) 8Khz, 8 bits mono	Rcv/Tx	Rcv/Tx	64 kb/s,
A	G.723 mono	Rcv	Rcv/Tx	6.5 kb/s
A	4-bit mono DVI, 8 Khz	Rcv/Tx	Rcv/Tx	32 kb/s
A	4-bit mono DVI 11.025 Khz	Rcv/Tx	Rcv/Tx	44 kb/s
A	4-bit mono DVI 22.05 Khz	Rcv/Tx	Rcv/Tx	Varies 0.0 -100 kb/s
A	MPEG Layer I 48 Khz @ 16 bits per sample, mono,			64 kb/s
A	MPEG Layer II 22 Khz @ 16 bits per sample, mono,			32 kb/s
A	MPEG Layer III 44 Khz @ 16 bits per sample mono	Rcv/Tx	Rcv/Tx	64 kb/s
V	JPEG (411, 422, 111) *	Rcv	Rcv/Tx	1.5Mb/s @ 320x200 @ 19fps
V	H.261	-	Rcv	-
V	H.263 **	Mode A Only	Rcv/Tx	120kb/s @ 176x144 @ 25fps
V	MPEG-I ***	Tx	Rcv/Tx	Depends on encoded media

** Video dimensions must be multiples of 8 pixels*
*** Can only be transmitted as one of 128x96, 176x144, 352x288*
**** Only from pre-encoded media e.g. mpeg encoded file.*

Table 1. JMF Supported Multicast RTP payloads

The primary channel of communication in a video conference is the audio channel, with the video link providing additional information, such as the expression of the other person. This generally aids in communication by providing telepresence and leads to fewer misunderstandings, even when the bit rate used for audio is low. The default native H.263 video codec has a relatively low bit rate and high picture quality.

4 Quality of Service Issues

An important distinction between different types of traffic is whether timeliness is more important than reliability. For example, a multi-way video-conference can tolerate the loss of some data in transit, but will run into severe difficulties if packets are not delivered within a given time frame. The reference application for the work described in this paper is to support synchronous communication between small groups of people using a variety of media. For example, in a tutorial session consisting of a tutor and half a dozen students the allocated resources would include synchronous duplex video and audio channels, and shared objects. Shared objects typically require full data integrity rather than very low latency and high bandwidth. On the web this usually means employing HTTP/TCP.

Groupware Environment *multiple resources with different QoS requirements*	
shared objects, usually server based	peer-to-peer multicast media flows
Hyper Text Transfer Protocol (HTTP)	Real Time Protocol (RTP) *a time-aware protocol*
Transmission Control Protocol (TCP) *two-party completely reliable data exchange, no time guarantees*	User Datagram Protocol (UDP) *adds port numbers to IP datagrams, the basis for best effort IP multicast*
Internet Protocol (IP) *unreliable, best effort*	

Table 2. Network protocols and the QoS they provide

Table 2 summarises the network protocols involved in supporting web-based groupware that includes real-time multimedia. A video connection using the JMF H.263 video codec at 176x144 and 24 f/s requires 120Kb/s and an audio connection using LPC encoding at 8 Khz mono requires 5.6Kb/s. per second. Putting these together means that a reasonable quality audio/video conference requires a total of 125.6Kb/s per sender. Multicast networking avoids the $2*n^2$ connection problem, where n is the number of participants. So, where a tutorial group consists of 6 students and one tutor, the overall bandwidth required is about 879.2Kb/s. The networks used to facilitate communication may have different characteristics for different groups. Postgraduates may engage in a small symposium, whilst being connected directly to 100Mb/s ethernet switches. An undergraduate tutorial might involve a shared connection to a 10Mb/s hub for participants connecting from student residences. The need may arise for cross-institutional conferences during which data must traverse each institution's connection to a wide area network, at which point competing traffic may severely constrain bandwidth availability and increase jitter. Some participants may participate from a home computer. Here the technology used in the "last mile" will constrain bandwidth. ADSL allows up to 256 Kb/s upload, ISDN-2 128 Kb/s and a V90 modem 56 Kb/s. In the scenario where bandwidth is plentiful, the system should be able to make full use of the available resources, thereby ensuring a good QoS for the participants. On the other hand, when bandwidth is scarce, a solution is required that does not waste the constrained resource or act unfairly in relation to competing traffic.

4.1 Case Study: The Finesse Groupware Environment

Finesse [9] is a TAGS-based learning environment used to teach fund management to Business and Finance students. At its core is a Portfolio Management Facility [10] which uses real time data[2] from the London Stock Exchange. Groups of students are allocated a Portfolio with a starting balance of £100 million. They can inspect changes and historic data in a range of companies, buy and sell shares, look at their portfolio profit and loss, and follow links to other sources of market information. Money left unused is credited with interest based on rates that are downloaded daily. Each group is also allocated a Notebook facility, to communicate with each other and their tutor concerning their investment decisions. The QoS for the share data is interesting in that it has to be both timely *and* reliable. Ideally, teams would also be allocated video conferencing sessions to further increase the real world dimension of the learning environment.

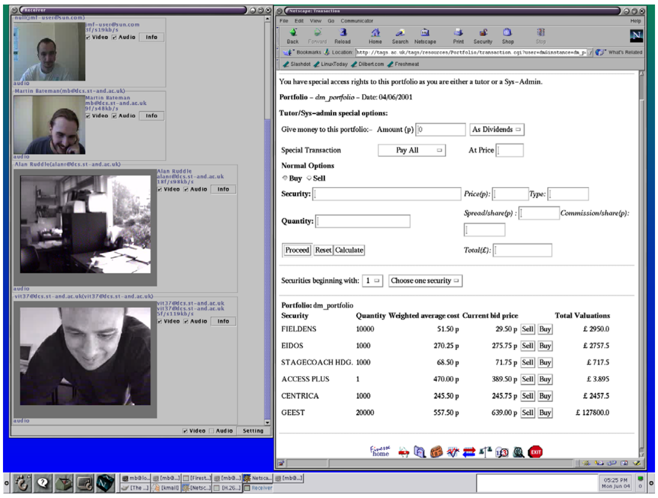

Fig. 2: A Finesse Session

An early attempt to do so using the Mbone tools co-located on the same machine as the web browser suffered from the same problems as the Warp-based shared spreadsheet. JMF-based applets makes it possible to allocate conference sessions to groups, in the same way that they are allocated Portfolios and Notebooks. Fig. 2 shows the Finesse environment (the transaction page is showing) expanded to include an allocated conference resource.

5 Adaptation Strategy

Research [11] has shown that consistency of quality is often more important to users than the actual quality of service achieved. For example, if during the lifetime of a video session, the achievable frame rate varies between two bounds, it is arguably

[2] The data source used is free, is 20 minutes old and is sampled every 5 minutes.

better to present using the lower bound throughout the session rather than try to dynamically optimise the quality. Given that it is desirable to avoid adaptation to network conditions during the lifetime of a session, it follows that a long period of probing from a low starting point for an appropriate operating point is undesirable. Previously, it has been envisaged that this problem could be addressed by combining admission control with resource reservation [12] to provide a guaranteed QoS for the duration of a session. Assuming (realistically) that such facilities are unavailable, what can be achieved in their absence?

The approach under investigation involves collecting statistics of the conditions experienced by past sessions in a common repository and then making predictions at the start of a conference about the conditions that are likely to prevail during its lifetime. This approach is similar to that adopted for a Location Information Server [13]. These predictions can be used to initialise relevant parameters, so that a QoS appropriate to the available network resources can be achieved at the start of, and maintained for the duration of, the conference. Furthermore this can be achieved without expert user intervention, which is an important consideration where the participants in a conference may not be expert users.

6 QoS Aware Conference Control

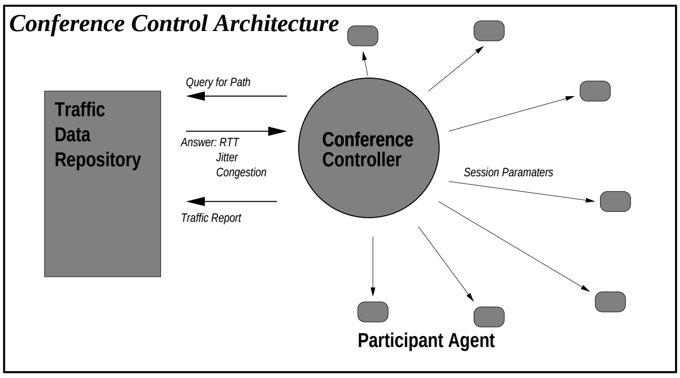

Fig. 3. Conference Control Architecture

There are three main components of the system shown in Fig.3; a Traffic Data Repository (TDR), a Conference Controller (CC) and Participant Agents (PAs). These components build upon the services provided by the Real Time Protocol (RTP) and the Real Time Control Protocol (RTCP).

RTP provides end-to-end network transport functions suitable for applications transmitting real-time data, such as audio, video or simulation data, over multicast or unicast network services. RTP does not address resource reservation and does not guarantee quality-of-service for real-time services. The data transport is augmented by a control protocol (RTCP) to allow monitoring of the data delivery in a manner scalable to large multicast networks, and to provide minimal control and identification functionality. An RTP session provides the association between a set of participants that are communicating using RTP. For each of the participants the session is defined by a network address and two ports. One port is for data and the other for control traffic. In a multimedia conference, each medium is carried in a separate RTP session with is own control packets. Thus in the scenario of a conference utilising video, voice, chat, a whiteboard and a presentation application there would be five RTP sessions. The session is also the granularity at which information about the state of the network is collected and disseminated to participants. In a session there may be multiple senders and receivers of data. RTCP Sender Reports contain a Synchronisation Source identifier (SSI), which is unique within the scope of a session. Associated with each SSI are a number of traffic statistics. These include timestamps to facilitate the calculation of Round Trip Times (RTTs), counts of the total packets sent and the total number of packets lost during the session, and the proportion of packets lost since the last report. There is also an estimate of the inter-arrival jitter.

6.1 Participant Agent

A PA is located on the server running the web environment. At start up it receives a description of the path to each participant from the conference controller and the sessions available in the conference. Based upon the bandwidth information the agent decides which sessions to subscribe to and, based upon the jitter description, it configures the size of the play out buffers for the subscribed sessions. The play-out buffers must be either set to an appropriately small size in order to maintain the real-time interactive nature of the system. There are situations where the play-out buffer could be increased, for example in a lecturing scenario. During the lifetime of the conference the PA receives RTCP Receivers and Senders reports. These reports contain estimates of packet loss and jitter for the media sources. The agent maps these statistics from SSIs to IP numbers. At the end of the conference and periodically throughout the conference traffic statistic reports are generated and sent to the conference controller.

6.2 Conference Controller

The CC is located on a central machine. At the initiation of the conference it queries the TDR for predictions of traffic conditions on the paths between conference participants. Based upon this information, it determines which media to use in the conference and the total bandwidth available on each path. From this information it uses a policy to partition the bandwidth between the available media and determines appropriate frame rates, resolutions and sampling rates. These policies can also take the relative importance of the users into consideration. For example, in a tutorial session the tutor's media could be given higher bandwidth allocations and therefore higher quality. For the duration of the conference, the CC monitors the entry and exit of participants and receives reports from PAs of the network conditions. In the case of a strong mismatch between expected and experienced conditions, it may be necessary to adjust bandwidth allocation and other parameters. When this happens, update instructions are sent to each PA, using the Lightweight Reliable Multicast Protocol. At the end of the conference, the CC generates a report to the TDR of the network conditions and traffic characteristics between each of the participants. This report allows the repository to determine the accuracy of its predictions and to update the data held on the utilised paths.

6.3 Traffic Data Repository

The TDR provides a central repository for information about network paths. Consequently, information gathered from one conference can be made available to future conferences. The age of the data held in the repository is also maintained. This enables a CC to make a judgement about the reliability of the data. The TDR provides an interface that allows it to be remotely queried by a CC or other application about the expected traffic characteristics for a path. The TDR will also receive reports from the CC about the network conditions experienced during a session.

6.4 Collecting the Path Characteristics

In the above architecture it is the responsibility of the Participant Agent to monitor RTCP reports and generate reports to be utilised by the CC. This involves two functions: determining the end point IP addresses of sources and receivers and extracting meaningful traffic statistics from RTCP Sender and Receiver Reports. These statistics are then aggregated into reports for the CC. The CC is then able to take any necessary control actions and generate its own reports to the TDR. RTCP packets contain reports for synchronisation sources and receivers. It is necessary to establish a mapping from the SSI to unicast IP addresses, because the SSI may change between sessions. This can be achieved by observing the IP address of the source. The IP address of the report originator can also be determined from the source IP address on the reports. With these addresses established statistics generated by RTCP can be associated with the path between the two unicast IP addresses. Traffic statistics need to be extracted from the data streams.

There are three main statistics of interest. The proportion of packets lost (congestion), the round trip time and the inter-arrival jitter.

- Congestion: Each receiver's report contains the fraction of packets lost since the previous report. The receiver calculates this fraction by dividing, the number of packets it received by the number of packets expected, since the last report.
- Round Trip Time Estimation: It is necessary to determine the RTT to facilitate mapping between fair window sizes and fair rates. Each sender's report contains an NTP time stamp for the time that the report was generated. This is used with receiver's reports to facilitate the calculation of RTTs.
- Jitter: It is necessary to determine the amount of jitter to configure the play-out buffer for real-time media. Too large a buffer will increase delay, too small a buffer will result in packets arriving too late to be of use. A measurement of inter-arrival jitter is included in receiver's reports. This is an estimate of the statistical variance of the RTP data packet inter-arrival time.

7 A Role for Avatars in Adaptation?

When advice from the CC suggests realtime video is not easily sustainable a model-based approach may prove useful in order to maintain telepresence. If a model of a participant is already present on each participant's machine then relatively low bandwidth control data can be multicast to animate that model. The need for a video stream is removed while the illusion of video conferencing still remains. From another perspective, if the quality of the video is likely to be very poor then an avatar may perform as well, or better, and use considerably less bandwidth.

Avatars may be created on the web by using the virtual reality modelling language (VRML) [14], the current standard being VRML97 [15], with X3D [16] currently being specified. A VRML viewer is used to view an avatar. These come in a number of different forms, applets/browser plug-ins, standalone applications and graphical components that can be embedded into other applications. A VRML viewer which conforms to the External Authoring Interface (EAI) [17] allows other applications to control what is happening within a virtual world. This is useful for conferencing as it allows the avatar to be synchronised with the audio channel (lip movements for example). This mechanism also provides a method of animating the avatar so that it moves like a desktop video conference participant, for example blinking and moving, instead of staying in the same place all the time.

Fig.4a. The basis for a simple avatar

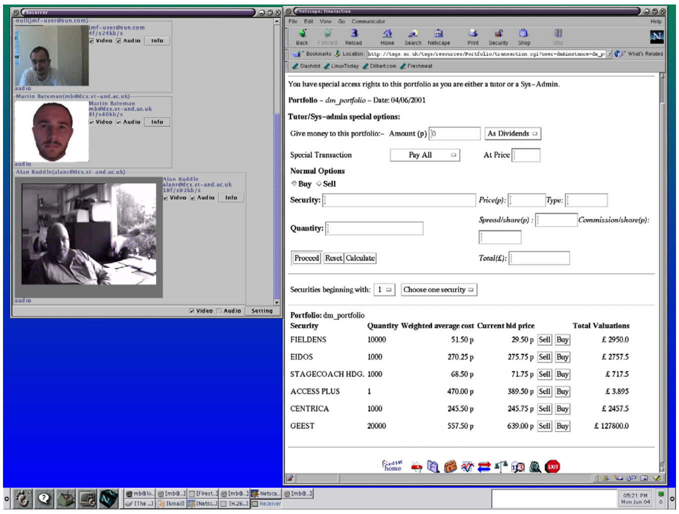

Fig.4b. The avatar in a conference with real actors

A simple avatar can be constructed from two perspectives and is around 80KB (Fig. 4a). This must be initially downloaded to each participant agent. In the case of the tutorial with 6 students and 1 tutor this would require 560KB of avatar data to represent all participants. With the use of silence suppression and avatars the bandwidth required to participate in a conference could be reduced from 880Kb/s per second to 6.Kb/s, (the bandwidth required for a single audio channel), assuming that only one person talks at a time. Figure 4b shows the avatar appearing in a video conference.

7.1 Face Tracking

Face tracking techniques [18] based on the position and direction of a participants head can be derived and used to control their avatar, thus returning some of the feedback that a standard video channel provides. A face tracking avatar requires 5 pieces of control information: the x, y, z positions in space and the horizontal and vertical roll. With this information the face can be placed in 3 dimensional space and the direction it is facing controlled. Control information is 75 bytes per message (assuming 16 bit number for each element), which at 16 frames per second is around 1K per second per person within the multimedia conferencing session. So, in the example tutorial session, this is only 7Kb/s per second for all the members of the session. However, facial tracking and image manipulation are potentially difficult to implement well and further work is necessary to evaluate the feasibility of this approach.

8 Conclusion

Although video conferencing is seen as an essential part of distributed groupware there is still little sign of its presence in the growing number of web-based groupware environments. When it is present it is not integrated with the web and is typically difficult to install and use. The JMF, which supports RTP, offers the possibility of creating video conferencing session applets which can be allocated using the same high-level mechanisms as other shared resources. An approach based on the use of TAGS groupware resource allocation system to incorporate RTP sessions implemented as JMF applets looks promising.

Inevitably, as with all time-sensitive media, QoS issues surface. It was noted that consistency throughout a session is preferable to chopping and changing with the prevailing network conditions. This raises the problem of finding an appropriate set of QoS initialisation parameters that can be realistically supported for the duration of a session. A Conference Control Architecture has been described that addresses this problem through the maintenance and use of a Traffic Data Repository to store and predict network conditions. The CCA also allows for different QoS initialisations for heterogenous connections. In much video conferencing the most important dimension is the audio, with the video providing realism through telepresence. An approach using the web-oriented VRML to create simple models of participant's faces - avatars - is being investigated as it has the potential to dramatically reduce bandwidth requirements for situations where that resource is scarce.

The systems described in this paper represent a novel and promising approach to realising real time multimedia groupware on the web. Further work will concentrate on their continued development and evaluation.

9 References

1. Allison, C., F. Huang, and M.J. Livesey, *Object Coherence in Distributed Interaction*, in *Multimedia'99*, N. Correia, T. Chambel, and G. Davenport, Editors. 1999, Springer-Verlag: New York. p. 123-132.
2. Allison, C., M. Bramley, R. Michaelson, and J. Serrano. *An Integrated Framework for Distributed Learning Environments*. in *"Advances in Concurrent Engineering", 6th ISPE International Conference on Concurrent Engineering*. 1999. Bath.
3. Allison, C., D. McKechan, A. Ruddle, and R. Michaelson. *A Group Based System for Group Based Learning*. in *European Perspectives on Computer-supported Collaborative Learning, the proceedings of Euro CSCL 2001*. 2001. Maastricht: Maastricht McLuhan Institute.
4. McCanne, S. and V. Jacobson. *VIC: A Flexible Framework for Packet Video*. in *ACM Multimedia*. 1995. San Francisco.
5. Perkins, C.S., O. Hodson, and V. Hardman, *A Survey of Packet-Loss Recovery Techniques for Streaming Audio.* IEEE Network, 1998.
6. Schulzrinne, H. and e. al., *IETF RFC 1889 RTP: a transport protocol for real time applications*. 1996.
7. Jacobson, V., *et al. A Reliable Multicast Framework for Lightweight Sessions and Application-Level Framing*. in *ACM SIGComm 95*. 1995.
8. Sun Microsystems. *JMF: The Java Media Framework*. 2001.
9. Helliar, C.V., R. Michaelson, D.M. Power, and C.D. Sinclair, *Using a Portfolio Management Game (FINESSE) to Teach Finance.* Accounting Education, 2000. **9**(1): p. 37-51.
10. Power, D.M., R. Michaelson, and C. Allison. *The Finesse Portfolio Management Facility*. in *9th CTI-AFM Conference*. 1998. York: CTI-AFM.
11. Bouch, A. and M.A. Sasse. *The Case for Predictable Network Service*. in *MMCN'2000*. 2000. San Jose: ACM.
12. Braden, B., D. Zhang, S. Estrin, and S. Jamin, *Resource ReSerVation Protocol (RSVP) -Version 1 Functional Specification*. 1996: IETF Working Document.
13. Ruddle, A. *A Location Information Server for the Internet*. in *International Conference on Communications and Computer Networks*. 2000. Las Vegas.
14. Web3D_Consortium, *VRML: The Virtual Reality Modeling Language*, in *http://www.vrml.org/*. 2000.
15. VRML_Consortium_Inc., *Part 1: Functional specification and UTF-8 encoding, International Standard ISO/IEC 14772-1*. 1997.
16. Web3D_Consortium, *X3D, http://www.web3d.org/TaskGroups/x3d/index.html*.
17. VRML_Consortium_Inc., *The virtual Reality Modeling Language (VRML) -- Part 2: External authoring interface, International Standard ISO/IEC FDIS 14772-2*. 2001.
18. Jie, R.S., U. Meier, and A. Waibel. *Real-time face and facial feature tracking and applications*. in *AVSP'98*. 1998. Terrigal, Australia.