Using the Java Media Framework to build Adaptive Multimedia Groupware Applications

Martin Bateman, Alan Ruddle, Colin Allison

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

Abstract

Realtime 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, and this is in part due to their implications for resource allocation and management. The Java Media Framework offers a promising means of redressing this situation. This paper describes an architecture for 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.

1 Introduction

Multimedia conferencing tools are specific applications within the general scope of real-time distributed groupware (RDG). Ellis and Gibbs [1] itemised the following requirements of RDG systems:

- response times must be short
- participants are not required to be connected to the same machine
- participants are free to come and go at any time
- participants are not required to follow a pre-planned script
- there is the possibility of a high degree of access conflict
- *participants can communicate with each other via an audio-visual channel*

Warp [2] produced an exemplar RDG application in the form of a multi-user, distributed, shared spreadsheet [3] that meets all of the above requirements. (See Fig. 1). However, a new important requirement must be added to the RDG requirements list, namely that RDG, in the interests of interoperability, should be *usable on the web*. 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 of implementation and deployment.

Figure 1: A Warp-based Multimedia Groupware Application

The Warp-based RDG application shown in Fig.1. used X-windows, IP Multicast, and RPC2, and is therefore an example of pre-web groupware. It is also a 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.

The TAGS framework for groupware [4] differs from Warp in that it is completely web-oriented. TAGS is a project to research the development and deployment of distributed learning environments and is currently used to support the development of groupware in the form of distributed, multi-user resources. 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 individual belongs to. It has however proved difficult to incorporate realtime audio and video conferences into this model, due in part to the general lack of mechanisms for integrating realtime conferences into the Web. Efforts to date have resorted to starting applications such as vic[5] and rat[6] 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 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 these problems in the form of an architecture for synchronous communication *within* a group-based distributed learning environment. Given the special needs of such resources the architecture provides for adaptation to predicted and current network conditions. In the broader sense it is an application of the principle that distributed learning environments should be QoS aware in order to provide a reliable service.

The remainder of the paper is structured into three sections. Firstly the Java Media Framework (JMF) and its features that facilitate web based synchronous communication are described and the limitations imposed by such an approach are identified. Secondly, the issues that arise when the JMF is used within the context of a system that supports the allocation of resources to users through a group mechanism are discussed. Finally, the automatic configuration of multimedia resource instances based on past traffic measurements to ensure fairness with competing traffic and utilisation of available bandwidth is discussed.

2 The Java Media Framework

The Java Media Framework (JMF) [7] 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 quite limited in ability since it is unable to capture or transmit video/audio. It can however, render audio/video streams that it receives and is intended for use in situations where the native version of JMF has not been installed but the user still wishes to receive the audio/video streams. In the native library version of the JMF the majority of the processing is performed in platform specific code. This has significant performance advantages over a pure Java implementation and offers greater capabilities for the programmer and user.

The simple JMF application shown in Fig.2 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 to the network.

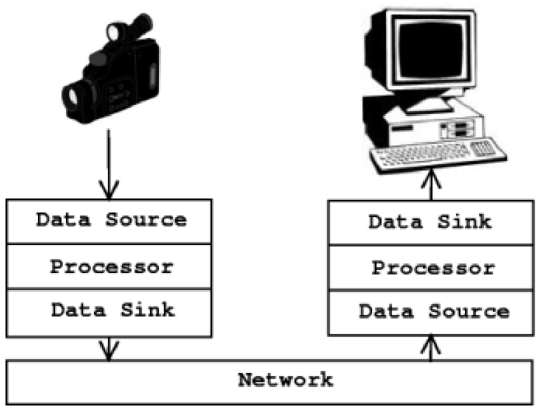

Figure 2: A simple JMF application

JMF supports various network protocols for the transmission of multimedia, the most interesting of these from this projects point of view is multicast RTP[8] which can be used to disseminate realtime multimedia to a group of users.

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

Table 1 shows the RTP payload types that are supported by the default installation of the JMF. Due to the lack of H.261 encoding support, JMF is only able to interoperate with vic using Motion JPEG, which is bandwidth hungry in comparison to H.261 or H.263.

JMF's major advantage when compared to other video conferencing technologies is that since it is Java based, it can be incorporated in applets that can then be integrated into a web environment. Additional multiplexers, demultiplexers, filters and codecs can be added to the JMF via means of a plug-in architecture. These plug-ins must be registered on the host machine before they can be used.

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, changing media.

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 and leads to fewer misunderstandings, even when the bit rate used for audio is low.

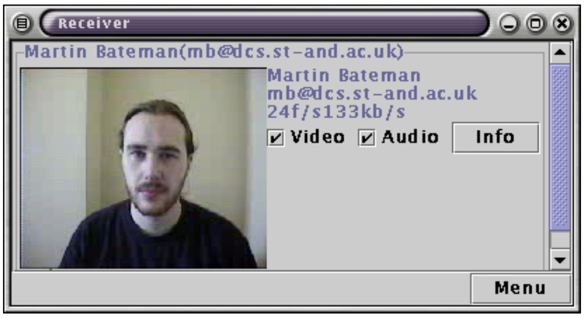

Figure 3: JMF Video Conferencing application

The prototype JMF based video conferencing application uses the architecture shown in Figure 3. By default it uses the native H.263 codec for video. This has a relatively low bit rate and high picture quality. There is also scope for the application to use Motion JPEG for the video transmission which enables it to interact with default installations of vic. The application is available as a Java Applet that can be included in Web pages.

3 Groupware Resource Allocation

TAGS has three main abstractions; 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.

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 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. Figure 4 illustrates an example set of relationships between users, groups and resources.

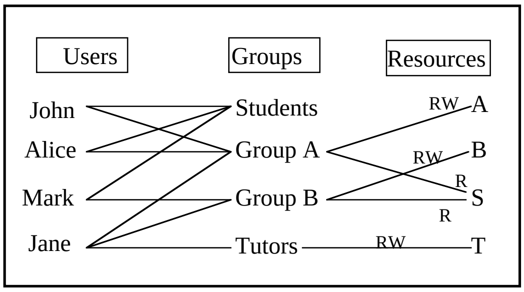

Figure 4: Group Resource Allocation

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 on a Read-only basis, whereas A is Read-Write. Mark and Jane are in Group B, which allows resource B as Read-Write and S as Read-only.

Could this allocation model be used to manage video conferences, as shared resources? Multicast RTP supports *sessions* and participants, where a session is a group of users. Sessions are short-lived entities in which the membership is decided by the members of the session. Anyone is free to enter or leave at anytime whereas a group is a long term entity, typically lasting around 10 weeks.

In order to interoperate with the TAGS framework, there must be functionality to allocate the sessions to groups from within TAGS. This would have the advantage that it would mean that multimedia sessions are easy to 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[9] 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 TTL, 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, but it does not eliminate the need for the session directory protocol. As stated, many sessions are already advertised within sdr and this information can be used when allocating session addresses so that a clash does not occur.

There are various security issues that should be addressed when using RTP multicast based multimedia conferencing. There is no inherent security built into RTP based multicast. However, the scoping of the multicast session (the Time To Live value) does provide a limited form of security in that you have to be within the scope of the multicast session in order to transmit or receive the RTP session. There is also an element of security through ignorance since in order to receive a multicast session the multicast address must be known.

4 Quality of Service Issues

The aim of this exercise is to support synchronous communication between small groups of people using a selection of mediums. For example a tutorial session might include a tutor and half a dozen students. Supported communication channels might include video, audio, and shared objects. A standard video connection using the H.263 video codec at 176x144 and 24 f/s requires 120Kb/s where as an audio connection using LPC encoding at 8 Khz mono requires 5.6Kb/s. Putting these together means that for a reasonable audio/video conferencing requires a total of 125.6Kb/s per sender in a conference. So, assuming that a tutorial group consists of 6 students and one tutor, the overall bandwidth required is 879.2Kb/s. The networks used to facilitate communication may have different characteristics for different groups. This can be illustrated by briefly considering a number of scenarios. Postgraduate students may engage in a small symposium, whilst being connected directly to 100 Mb/s ethernet switches. An undergraduate tutorial might involve a shared connection to a 10 MBit/s hub for many of the participants, as well as connections from student residences. The need may arise for cross-institutional conferences during which data must traverse each institutions connection to a wide area network, at which point competing traffic may severely constrain bandwidth availability and increase jitter. Another scenario may involve some participants logging on to the conference 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 high Quality of Service for the participants. On the other hand, when bandwidth is seriously constrained, a workable solution is required that does not waste the constrained resource or act unfairly in relation to competing traffic.

5 Adaptation Strategy

Research [10] 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 framework varies between two bounds, it is arguably 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 [11] to provide a guaranteed Quality of Service 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 prevail during its lifetime. This approach is similar to that adopted for a Location Information Server [12]. These predictions can be used to initialise relevant parameters, so that a Quality of Service 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

There are three main components of the system; a Traffic Data Repository (TDR), a Conference Controller (CC) and Participant Agents (PAs). (See Fig.5).

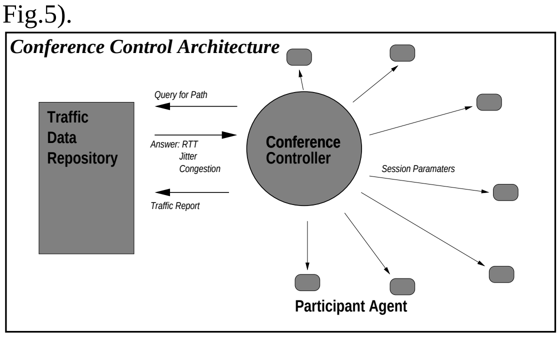

Figure 5: Conference Control Architecture

These components build upon the services provided by RTP and the Real Time Control

Protocol (RTCP). RTP is discussed before describing the function of each of the other components.

6.1 RTP and RTCP

RTP provides end-to-end network transport functions suitable for applications transmitting realtime 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 participant the session is identified 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 (SSRC), which is unique within the scope of a session. Associated with each SSRC are a number of traffic statistics. There are NTP 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.

6.2 Participant Agent

A PA is located on the server running the web environment. At start up, it receives from the conference controller, a description of the path to each participant and of 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 playout buffers must be set to an appropriately small size in order to maintain the realtime interactive nature of the system. There are situations where the playout 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 synchronization source identifiers 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.

6.3 Conference Controller

The CC is located on a central machine. At the initiation of the conference it queries the Traffic Data Repository (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, since what the tutor is saying is normally of increased importance. 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, updates instructions are sent to each PA, using the Lightweight Reliable Multicast Protocol (LRMP). 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.

6.4 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 judgment 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.

6.5 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 SSCR to unicast IP addresses, because Synchronisation Source identifier may change between sessions. This can be achieved by observing the IP address of the source. The IP address of the report originater 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 at least three statistics of interest. The proportion of packets lost (congestion), the round trip time and the inter-arrival jitter.

1. Congestion: Each receivers 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.
2. Round Trip Time Estimation: It is necessary to determine the RTT to facilitate mapping between fair window sizes and fair rates. Each Senders Report contains an NTP time stamp for the time that the report was generated. This is intended to be used with receivers reports to facilitate the calculation of RTTs. This enables an approximation of the RTT to be calculated.
3. Jitter: It is necessary to determine the amount of jitter to properly configure the play out buffer for realtime media. Too large a buffer will increase delay, too small a buffer will result in packets arriving after they should have been played thereby reducing the quality of audio and video playback. A measurement of inter arrival jitter is included in receivers reports. This is an estimate of the statistical variance of the RTP data packet inter arrival time, measured in time stamp units and expressed as an unsigned integer.

7 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 offers the possibility of creating video conferencing session applets which can be allocated using the same high-level mechanisms as other shared resources. The JMF supports RTP and the use of the TAGS 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. The Conference Control Architecture described 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.

8 References

1. Ellis, C.A., S.J. Gibbs, and R.G. L., *Groupware -- Some Issues and Experiences.* CACM, 1991. 34(1): p. 39-58.
2. Allison, C., P. Harrington, F. Huang, and M.J. Livesey. *A Global Atomicity Primitive.* in *"IWOOOS'96", 5th International Workshop on Object Orientation in Operating Systems.* 1996 Seattle: IEEE Press.
3. 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.
4. 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. 43-50 Maastricht: Maastricht McLuhan Institute.
5. McCanne, S. and V. Jacobson. *VIC: A Flexible Framework for Packet Video*. in *ACM Multimedia*. 1995. 511-522 San Francisco.
6. Perkins, C.S., O. Hodson, and V. Hardman, *A Survey of Packet-Loss Recovery Techniques for Streaming Audio.* IEEE Network, 1998.
7. Microsystems, S. *JMF: The Java Media Framework*. 2001. http://java.sun.com/products/java-media/jmf.
8. Schulzrinne, H. and e. al., *IETF RFC 1889 RTP: a transport protocol for real time applications*. 1996.
9. Jacobson, V., *et al. A Reliable Multicast Framework for Lightweight Sessions and Application-Level Framing*. in *ACM SIGComm 95*. 1995. 342-356.
10. Bouch, A. and M.A. Sasse. *The Case for Predictable Network Service*. in *MMCN'2000*. 2000. 188-195 San Jose: ACM.
11. Braden, B., D. Zhang, S. Estrin, and S. Jamin, *Resource ReSerVation Protocol (RSVP) -Version 1 Functional Specification*. 1996: IETF Working Document.
12. Ruddle, A. *A Location Information Server for the Internet*. in *International Conference on Communications and Computer Networks*. 2000 Las Vegas. IEEE.