For: casner
Printed on: Mon, Jul 27, 1992 20:21:17

Document: CCR-audiocast
Last saved on: Mon, Jul 27, 1992 20:21:50

ISI Reprint Series
ISI/RS–92–293
July 1992

Stephen Casner
Stephen Deering

First IETF Internet Audiocast

Reprinted from
ACM SIGCOMM
Computer Communications Review,
Vol.22, No.3 (July 1992).

University of Southern California
Information Sciences Institute
4676 Admiralty Way, Marina del Rey, CA 90292–6695
310–822–1511

An earlier version of this article appeared in *ConneXions, The Interoperability Report,* Vol.6, No.6 (June 1992).

This research was sponsored by the Defense Advanced Research Projeccts Agency under contract number DABT63–91–C–0001. Views and conclusions contained in this report are the authors' and should not be interpreted as representing the official opinion or policy of the U.S. Government or any person or agency connected with it.

Approved for public release; distribution is unlimited.

First IETF Internet Audiocast

Stephen Casner
USC/Information Sciences Institute
casner@isi.edu

Stephen Deering
Xerox Palo Alto Research Center
deering@parc.xerox.com

1 Introduction

At the March, 1992 meeting of the Internet Engineering Task Force (IETF) in San Diego, live audio from several sessions of the meeting was "audiocast" using multicast packet transmission from the IETF site over the Internet to participants at 20 sites on three continents spanning 16 time zones. This experiment was not only the first sizeable audio multicast over a packet network, but also significant for the size of the IP multicast network topology itself.

The audiocast included all the general sessions plus a few working group breakout sessions. Unlike listening to a radio broadcast, the remote participants could also talk back, as was demonstrated during a brief technical presentation on the experiment. Though the audio transmission was not perfect, it worked well enough in both directions that remote participants were able to ask cogent questions and engage in the discussions during the working group sessions.

This event was a demonstration of technology developed and tested in the DARTnet research testbed network.[1] It was a pilot experiment that we hope will be expanded at future IETF meetings to reach more destinations and to include video, images, and "shared whiteboards" along with audio. This is a step toward a more distributed IETF, a goal Dave Farber and Jack Haverty challenged the community to pursue during a discussion on the IETF mailing list last fall.

2 Technology

Three key elements enabled the audiocast:

- Readily-available hardware and software to generate and receive audio packets at the endpoints.
- IP multicast routing [1] to replicate the packets efficiently for distribution to a large number of recipients.
- Real-time network performance, in this case achieved only by selecting uncongested networks with sufficient bandwidth.

[1]DARTnet and many of the researchers using it are supported by the Defense Advanced Research Projects Agency.

2.1 Packet audio hardware and software

Audio hardware is now built into many workstations, such as Sun and NeXT, and is ready for deployment of software. To simplify this pilot experiment, we kept to the same hardware and software already tested in DARTnet. Of four interoperable packet audio programs, three run on Sparcstations (`VT` from ISI, `vat` from Lawrence Berkeley Laboratory, and `NEVOT` from University of Massachusetts) and one runs on a 386 PC plus audio card (from MIT).

At the IETF site and most of the remote sites, we ran `vat`, the Visual Audio Tool, written by Van Jacobson and Steve McCanne. In an X window, `vat` displays VU meters and volume control sliders for the microphone and speaker levels plus a status display identifying the participants in the conference.

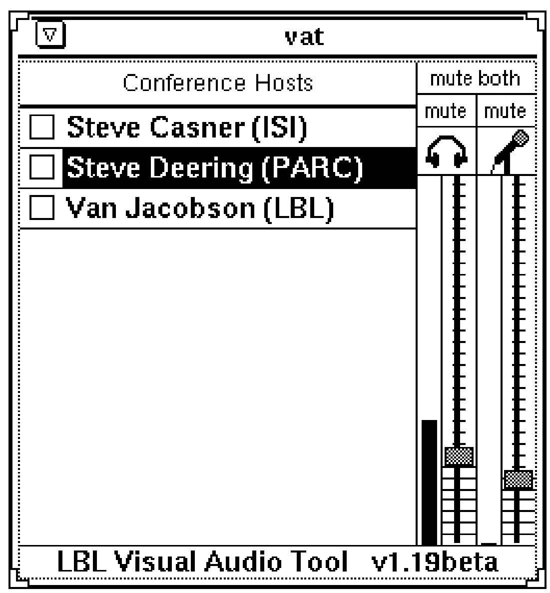

Figure 1: Image of a `vat` window

Behind the user interface, there are several important functions required to process audio packets:

- Silence/sound detection to avoid sending packets during silence, preferably with a dynamic threshold to accommodate varying levels of background noise.

- An adaptive playout delay to achieve continuous playback even though network delays vary. For multimedia systems, playout delays are also used to achieve synchronization among the media.
- Resequencing of out-of-order packets and filtering of duplicate packets within the playout delay buffer.
- Mixing of audio sample streams when packets arrive from multiple sites at the same time.
- A means to suppress acoustic feedback from the loudspeaker to the microphone that produces an echo at the far end (headphones are one solution, half-duplex speakerphone mode is another).

Algorithms for these functions have been developed over many years of research in packet audio, including early work [2,3] based on ARPAnet.

2.1.1 Voice data rates

For this IETF audiocast, we used the 64 Kb/s PCM audio produced by the Sparcstation /dev/audio device directly. Each packet contains 180 PCM voice samples, corresponding to an interval of 22.5 milliseconds and a rate of 44.4 packets per second. As shown in figure 2, the packet overhead is 32 bytes not counting any MAC header, resulting in a peak overall data rate of 75 Kb/s. However, since no packets are transmitted during silence, the average data rate is less. As software bandwidth compression algorithms are implemented in packet audio programs in the future, it will be possible to reduce the data rate for operation over slower network links.

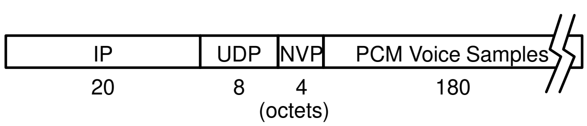

Figure 2: Audio packet format

2.1.2 Protocols

Note that packet audio is transported by UDP rather than TCP, for two reasons. First, TCP's reliability and flow control mechanisms aren't appropriate. Occasional packet loss causes only a small and acceptable reduction in audio quality, while allowing time for retransmission would require a longer playback delay, making interactive conversation more difficult. Flow control is not required because packets are generated at a regular rate and must be communicated and consumed at that rate. (Some systems may allow the data rate to be adjusted to compensate for network load.)

The second reason for choosing UDP is that it works well with IP multicast to reach many destinations, while TCP is limited (so far, at least) to point-to-point connections.

UDP lacks two functions of TCP that are needed: packet reordering and duplicate filtering. We add another protocol layer to provide these functions. As an interim convention, we use the data packet header from the Network Voice Protocol (NVP-II) [4], as shown in figure 3. For this version of PCM audio, the timestamp field increments every 22.5 millisecond packet interval, including during silence when no packets are transmitted. This provides sequencing and duplicate detection within a packet lifetime of about 20 seconds. The separate sequence number increments once per packet transmitted, enabling the detection of lost packets versus packets not transmitted during silence.

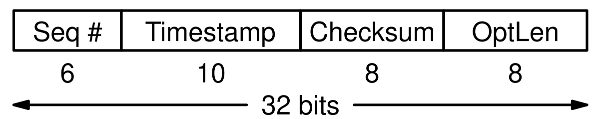

Figure 3: NVP header format

The NVP header is efficient (only 32 bits), but that makes some of the fields too small to support current requirements. It is in the charter of the Audio/Video Transport working group of the IETF to devise one or more replacement protocols for packet audio, video, and perhaps other media. At the San Diego meeting, a minimal strawman protocol was proposed, followed by a discussion of what functions should be added to it.

2.2 IP multicast routing

After end-system hardware and software for packet audio, the second key element required for the IETF audiocast was IP multicasting. For a large teleconference like this audiocast, the bandwidth and processing required for the source host to send a separate copy of each packet to each destination would be prohibitive. With IP multicast extensions implemented in the participating hosts and routers, the source can send a single copy of each packet which is then replicated as needed at each branching point on the logical tree reaching out to the destinations.

Unfortunately, few routers in the Internet today implement IP multicast routing. Fortunately, the experimental DARTnet routers do, using the Distance Vector Multicast Routing Protocol (DVMRP) implemented by Steve Deering in the `mrouted` daemon plus kernel extensions. This makes it easy for the DARTnet experimenters to hold teleconferences every week using packet audio and video. The idea of the audiocast experiment was to scale up to a larger number of destinations using DARTnet a transcontinental multicast backbone. However, just as we must reach some DARTnet experimenters who are not directly connected to DARTnet, for the audiocast we needed to extend far past DARTnet to the participating sites.

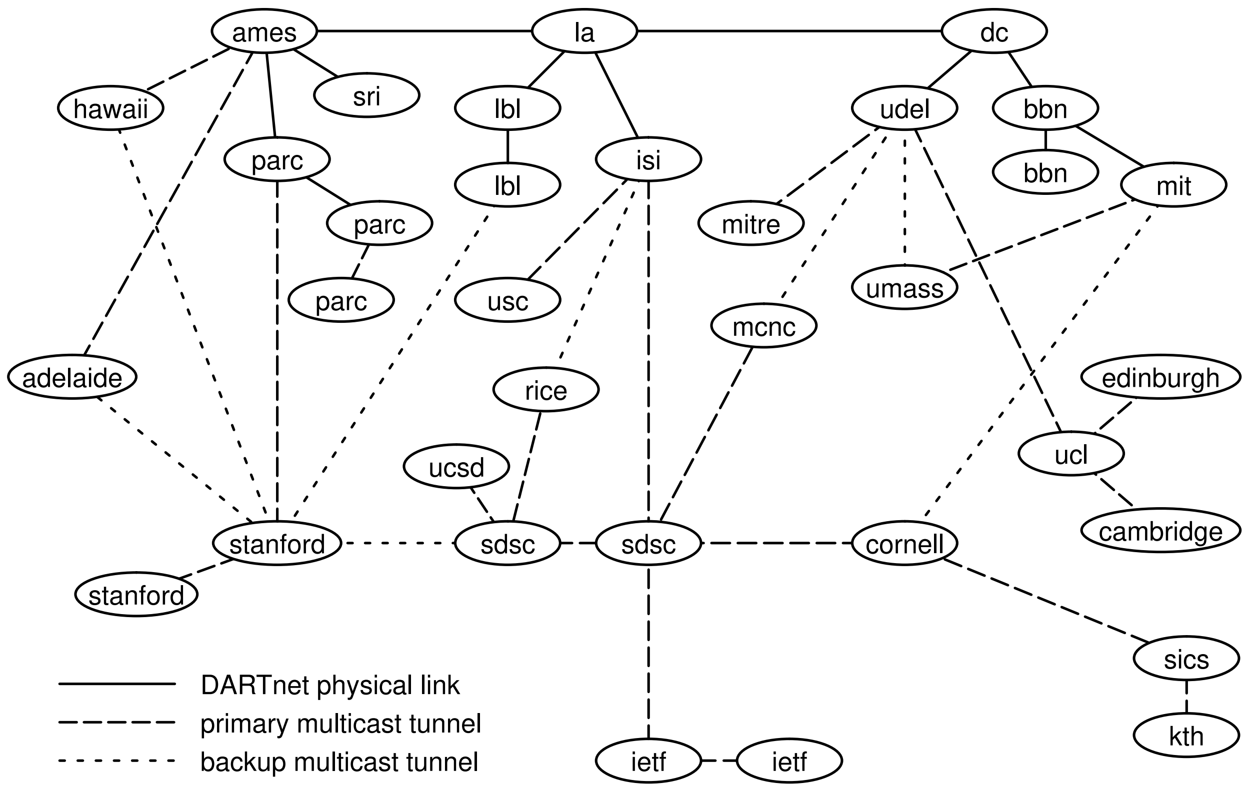

Figure 4: Multicast router topology for IETF audiocast 3/92

In order to support multicasting among subnets that are separated by (unicast) routers that do not support IP multicasting, `mrouted` includes support for "tunnels," which are virtual point-to-point links between pairs of `mrouted`s located anywhere in the internet. To transmit a multicast packet through a tunnel, a multicast router modifies the packet by appending an IP Loose Source Route option to the packet's IP header. The multicast destination address is moved into the source route, and the unicast address of the router at the far end of the tunnel is placed in the IP Destination Address field. Thus, the packet looks like a normal unicast packet to the routers and subnets along the path of the tunnel. The router at the far end of the tunnel restores the original multicast destination address and deletes the source route before forwarding the packet.

The tunnel mechanism allows `mrouted` to establish a virtual internet, for the purpose of multicasting only, that is independent of the physical internet, and that may span multiple administrative domains. For DARTnet experimenters' teleconferences, we use only a few tunnels extending from DARTnet nodes to individual sites, but for the audiocast we needed a whole network of tunnel links with DARTnet as a backbone. That network, shown in figure 4, set a record as the largest wide-area IP multicast network to date, with 34 routers linking 40 subnets.

The solid lines near the top of the graph indicate physical DARTnet links. The other lines are all multicast tunnels (virtual links) over a variety of Internet paths including the T3 NSFnet and international links. By assigning different metrics to different links, we established primary tunnels (dashed lines) and backup tunnels (dotted lines) to be used in case of failure in the primary tunnels or DARTnet lines.

2.3 Real-time network performance

IP multicast routing allows more efficient delivery of a packet to multiple destinations, but there may still be congestion on one or more links of the tree, causing delay or packet loss. Assuring timely delivery will require traffic control mechanisms in the network (see section 4.2.1). These don't exist yet, but there are a number of research efforts currently underway to solve this problem.

For this experiment, all we could do was to avoid congestion as much as possible by selecting network paths with sufficient bandwidth and low load levels.

3 What we learned

The IETF audiocast presented several problems not encountered in previous DARTnet teleconferences. Van Jacobson produced five new versions of `vat` during IETF week, adding new features to improve performance. For most of the audiocast, listener reports of the sound quality ranged, over time and place, from "very clear" to just "intelligible." Listeners were able to comprehend most of what was said, but there were several factors that caused dropouts in the playback.

The most obvious and expected cause of dropouts was that competing traffic would cause variance in the transit delay, especially on longer paths. To compensate, the delay adaptation algorithm in `vat` was improved and a "lecture mode" was added. This mode lets the playout delay remain large to minimize the number of packets declared late, for use during lectures when reduced interactivity is acceptable.

The second problem had nothing to do with packets, but with microphone placement and silence suppression. During the first working group session, we did not have a lapel microphone but attempted to pick up the presenters with a table microphone. The signal-to-noise ratio was too low because the presenters were too far from the microphone and there was too much ambient noise in a room divided by movable walls. A related problem was that, even in other sessions when the speaker used a lapel microphone, comments from the audience were not picked up well enough by the room microphones and were suppressed as silence. To avoid this problem, another control was implemented in `vat` to turn off silence suppression for use during presentations.

On the other hand, disabling silence suppression may exacerbate network loading and the resulting dropouts. Simon Hackett postulated that presenting a continuous load makes a non-trivial increase in the average bandwidth and eliminates the gaps that may give routers a chance to empty their queues. In addition, the packet audio receivers generally make playout delay adjustments during silence intervals, so drift between the sending and receiving clocks must be accommodated in some other way for continuous transmission.

The third cause of dropouts was a persistent packet loss estimated at 10-20% that we could not find nor explain. This loss was not evident in tests with `ping` and `traceroute`. Apparently there was no loss across the local T1 line connecting the source host in the main ballroom to the terminal room where another host running `vat` monitored the signal. We believe the loss occurred somewhere between the IETF terminal room and DARTnet, perhaps between IETF and SDSC. Even though the total traffic from the terminal room may have been hundreds of packets per second, that should not have created enough congestion to cause persistent, low-level packet loss.

One potential explanation would be that the multicast tunnel packets use the IP Loose Source Route option, which diverts those packets to a slower processing path in some router architectures. However, we also tested with source-routed test traffic at rates similar to the audio and did not see the same loss. In any case, there are plans to modify `mrouted` to encapsulate in a separate IP header rather than use the source route option.

We would likely have been able to find the cause of the packet loss if we had enough time to investigate fully. Clearly we need better measurement tools and procedures to accelerate the process. It would also have been very helpful if we had some way to monitor reception at a distance so we could tell when there were problems before the remote listeners contacted us by e-mail.

4 Looking into the future

4.1 Enhancements for IETF meetings

While the transmission of audio alone was interesting to the remote participants, in part because of its novelty, it is clear that the visual content of the presentations should also be provided. As one experiment, Ralph Droms made available via FTP the PostScript and ASCII versions of the transparencies before his talk on "Dynamic Host Configuration Protocol." The response was very positive.

Over DARTnet we regularly send packet video as well as packet audio. The data rate we typically use per video source is 128Kb/s, only twice the audio rate. It should be possible to implement software to decode that video and display it in an X window. This is under investigation at ISI. Other implementations are also feasible, for example some software encoding of video grabbed at a slow frame rate from a device such as the Sun VideoPix card.

Video of the presentations would give a better sense of what was happening, but video does not have adequate resolution to show slides clearly. For that, an automated slide presentation tool would be preferable, working from on-line source material or an on-site scanner. To generalize further for interactive working group discussions as well as presentations, a "shared whiteboard" onto which slides can be posted would be even better. Several research projects are working on such tools. Future IETF meetings would provide an excellent opportunity to test these tools on a large scale.

Perhaps the biggest impediment to adding visual media will not be network bandwidth or implementation of new tools, but logistics at the IETF site. It was difficult enough to connect into the ballroom audio system in such a way that a moderator could preview questions from remote sites before disturbing the local participants. Video cameras and people to operate them would add another logistical dimension and an additional expense that may not be feasible at this point.

Yet another level of complexity would be introduced if multiple working group meetings were to be teleconferences. Especially for the larger working groups, room acoustics would be a problem, both for picking up the voices of all participants and to play the sound from the remote end at sufficient volume and still cancel the echo caused by the microphones picking up that sound. Also, the resolution of compressed video is not sufficient for more than about six people per site. Rooms can be outfitted with equipment and video production personnel to handle these problems, but the cost is substantial. Meeting sites with a video-equipped auditorium and 20 breakout rooms equipped for videoconferencing are probably rare.

For meetings held in hotels, network connectivity within the hotel is another issue, especially if multiple rooms will be used for working group teleconferences. The aggregate bandwidth may also require more than a single T1 line from the IETF site to the Internet connection point. Beyond that point, the next hurdle is wide-area network support for significant levels of real-time traffic.

4.2 Scaling up to widespread packet audio

To scale up for widespread use of packet audio and video in multiple simultaneous public meetings, private teleconferences and other applications will require additional research and infrastructure engineering in several areas including network resource management, IP multicast routing and connection/session management.

4.2.1 Network resource management

The current work on protocols for packet audio and video transport over UDP is considered experimental because UDP transmission is only sufficient for small-scale use over fast portions of the Internet. If many people tried to send packet audio today, significant network congestion would likely result. Since packet audio does not practice congestion control, well-behaved TCP traffic would back off and let the audio take over which is probably not fair. Even so, when there is congestion, the resulting packet loss would impair the audio quality as well.

Research is underway on DARTnet and elsewhere to develop resource management (or traffic control) algorithms to solve this problem [5,6,7]. These algorithms, running in the various levels of packet switches in the network, would give priority to real-time traffic such as audio and video to achieve low delay and packet loss. At the same time, the algorithms would prevent real-time traffic from using more than its fair share, as determined by payment or policy. On small links with a low degree of multiplexing, new calls may be blocked when there is insufficient capacity to avoid degrading interference with established calls. The audio/video transport protocols may be used in conjunction with other protocols such as ST-II [8] or connectionless resource setup protocols to access the resource management functions.

4.2.2 Real IP multicast routing

A big part of the effort required to set up this IETF audiocast was in constructing the virtual network of multicast tunnels by hand and testing its performance. The tunnel mechanism in `mrouted` was only intended for experimental support of internet multicasting, pending widespread support for multicast routing by the regular routers. We need multicast routing in the real networks to reduce the setup effort and make the routing more robust.

To reach the goal of ubiquitous support for multicasting, more work on multicast routing protocols is needed. `Mrouted` suffers from the well-known scaling problems of any distance-vector routing protocol, and does not support hierarchical multicast routing or interoperation with other multicast routing protocols. Multicast extensions are being tested for the OPSF [9] and BGP [10] routing protocols, but complete coverage will likely require the design of a hierarchical scheme incorporating these and other protocols.

4.2.3 Control protocols

IP multicast addressing works well for broadcast-type applications such as this audiocast where *a priori* agreements on addressing and media encoding can be made. A predefined multicast address and UDP port number are built in to `vat`, so to join in one can just start listening. To expand from a single broadcast to multiple, private teleconferences, connection/session management protocols will be required to request call acceptance, negotiate compatible encodings, dynamically allocate IP multicast addresses, etc. Since IP multicast traffic may be received by anyone, the control protocols must handle authentication and key exchange so that the audio/video data can be encrypted.

Like resource management, connection management is the subject of current research. We expect that standards-track protocols integrating transport, resource management, and connection management will be the result of later IETF working group efforts.

5 Summary

The first IETF audiocast was an interesting and valuable experiment both for the experimenters and the participants. Though there were some problems, the results were good enough to suggest that the experiment be continued for future IETF meetings. There are several open issues that provide promising areas for additional work:

- better real-time performance measurement tools
- new application hardware (e.g., video cards), software (e.g., “shared whiteboards”), and protocols
- real-time traffic support (resource management)
- ubiquitous multicast routing support
- meeting site networking and studio facilities

Meanwhile, small-scale experiments with packet audio and video are encouraged in order to learn more about the protocol requirements. You can participate—see the appendix for details.

Acknowledgements

We thank Allison Mankin for suggesting the idea of the IETF audiocast. Many people helped to make the audiocast work. We particularly thank Paul Love, Tom Hutton and the rest of the SDSC crew for help getting the hardware and software set up at IETF; Steve Coya and Megan Davies for logistics; and Van Jacobson for work on `vat` and help in searching for the packet losses. Walt Prue set up a special route for the tunnel from San Diego to DARTnet at ISI, and Milo Medin and Jeff Burgan set up a connection from DARTnet to FIX-West for a clear shot to Hawaii and Australia.

We also thank the remote `vat` participants for their help as guinea pigs during testing. Simon Hackett in Adelaide had to get up very early in the morning; Ian Wakeman and Jon Crowcroft in London, along with Anders Klemets and Steve Pink in Stockholm, had to stay up late at night. This points out one impediment to "distributing the IETF" that will be hard to fix: too many time zones!

References

[1] S. Deering, *Host Extensions for IP Multicasting*, RFC 1112, August 1989.

[2] J. W. Forgie, "Speech Transmission in Packet-Switched Store and Forward Networks," *Proc. National Computer Conf.*, pp. 137-142, 1975.

[3] D. Cohen, "Issues in Transnet Packetized Voice Communication," *Proc. Fifth Data Communications Symposium*, pp. 6.10-13, September 1977.

[4] D. Cohen, "A Network Voice Protocol NVP-II," USC/Information Sciences Institute, April 1981.

[5] S. Floyd, "Issues in Flexible Resource Management for Datagram Networks," presented at *3rd Workshop on Very High Speed Networks*, March 1992.

[6] D. Clark, S. Shenker, L. Zhang, "Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanism," to appear in *Proc. SIGCOMM'92*, August 1992.

[7] D. Ferrari and D. Verma, "A Scheme for Real-Time Channel Establishment in Wide-Area Networks," *IEEE J. Sel. Areas in Comm.* SAC-8, April 1990.

[8] C. Topolcic, ed., *Experimental Internet Stream Protocol, Version 2 (ST-II)*, RFC 1190, October 1990.

[9] J. Moy, *OSPF version 2*, RFC 1247, July 1991.

[10] K. Lougheed and Y. Rekhter, *Border Gateway Protocol (BGP)*, June 1990.

Appendix

Software used in this experiment is available for testing by others who have Sun Sparcstations.

A pre-release of the LBL audio tool `vat` is available by anonymous FTP from ftp.ee.lbl.gov in the file vat.tar.Z. Included are a binary suitable for use on any version of Sparcstation and a manual entry. The authors of `vat` say the source will be released "soon."

In addition, a beta release of both binary and source for the UMass audio tool `NEVOT` is available by anonymous FTP from gaia.cs.umass.edu in pub/nevot-0.91.tar.Z (the version number may change).

You can test `vat` or `NEVOT` point-to-point between two hosts with a standard SunOS kernel, but to conference with multiple sites you will need a kernel with IP multicast support added. IP multicast invokes Ethernet multicast to reach hosts on the same subnet; to link multiple subnets you can set up tunnels, assuming sufficient bandwidth exists.

You don't need kernel sources to add multicast support. Pick up the file vmtp-ip/ipmulti-sunos411.tar.Z by anonymous FTP from gregorio.stanford.edu. It contains the IP multicast code to be added to a SunOS 4.1.1 kernel. Once you build the kernel, you should use `adb` to permanently patch the kernel variable audio_79C30_bsize from the standard value of 1024 to be 180 decimal to match the audio packet size for minimum delay. Otherwise there will be bad breakup when sound from two sites gets mixed for playback.

If you don't have a microphone for your Sparcstation, you can get one from Sun (part number 370-1414) or you can pick up an inexpensive microphone from Radio Shack or other suppliers. Walkman-style headphones are also recommended to allow full-duplex conversation.

Send a message to rem-conf-request@es.net to join the discussions of the IETF Audio/Video Transport and Remote Conferencing Architecture working groups.