PROVIDING MULTIMEDIA FACILITIES IN A WORKSTATION INDEPENDENT FORM

Mark F. Schulz and Lenard J. Payne

Intelligent Machines Laboratory
Department of Electrical Engineering
University of Queensland, Australia

ABSTRACT – This paper presents the planned work on providing multimedia facilities to a community of users of workstations and PCs. We aim to provide multimedia servers on a network, at least one per workstation, with each server providing a small number of services to the user. The workstation communicates with the multimedia server over an Ethernet via standard TCP/IP protocols. By providing the services separate from the workstation, we are able to upgrade workstations and network services with minimum interruption to our multimedia service.

INTRODUCTION

Multimedia applications have not yet been successfully merged into the workstation marketplace, although a small number of vendor specific attempts have been made. There are a number of factors that have contributed to this state of affairs, with the result that there is no existing link between the world of telecommunications and the world of the computer workstation. The introduction of ISDN will have some impact in this area, but to what extent is still largely unknown.

This project aims to provide a wide range of multimedia services to a heterogeneous community of workstations and PCs. Initially, a large portion of our work will be concerned with the provision of audio services. These audio services range over links to standard telephone lines, support for server-to-server voice communication over the local area network, and hybrid versions of these facilities such as automatic call routing from a telephone to a multimedia server, etc. We discuss proposed applications in a later section of this paper.

One must also distinguish between video services and other services. Because of the bandwidth required to transmit video images, it is not feasible to isolate video services in the manner we propose here. In the case of video, it is still necessary (at the moment) to transmit the video signal directly to a video card installed in the workstation, and to have this card handle the display of the video data. However, there is no technical reason that the selection of the source of the video data cannot be controlled (i.e., switched) by the multimedia server proposed here.

There also remains the wider problem of integrating multimedia into the user interface, e.g., making sound a resource that can be manipulated just like any other resource. This is a longer range project and is not addressed in this paper. It is being actively researched at M.I.T. (Schmandt and Arons [1984]), Olivetti (Arons et al. [1989]), and Apple (Ambron and Hooper [1990]).

EXISTING MULTIMEDIA SERVERS

There would appear to be no existing work which is quite as ambitious as that proposed here. There has been some research into providing audio services to users, and it is this reasearch which we examine in this section.

A simple audio server was developed at the M.I.T. Media Lab (Schmandt and Arons [1984]). An IBM XT containing commercially available voice and telephone cards from a number of vendors was connected to a telephone line. An audio server was then connected to the XT via a slow speed serial line. Commands to control the XT were passed over this serial line. Low level functions supported by the server included record/playback, editing, telephone interface, text-to-speech synthesis, and speech recognition. Only one user could access the server at a time, and only one server was ever assembled since the server-to-server access mentioned above was not envisaged.

Xerox PARC developed the Etherphone system (Zellweger, Terry and Swinehart [1988]) (Terry and Swinehart [1987]) which uses a centralized voice storage server. Conversations take place over the Ethernet, rather than over the telephone lines. The Etherphone project successfully integrated audio into the Cedar (Teitelman [1984]) programming environment (a Xerox PARC developed distributed operating system), making voice accessible from text editors and other applications. However, there was less emphasis on demanding user-interface functions than in the Media Lab voice server.

The Olivetti VOX Audio Server (Arons [1990]) is a descendent of the M.I.T. project. The aim of this project is to provide a kit of interactive audio building blocks for the easy integration of audio into a workstation. Initially, workstations will be equipped with a mixer, microphones, speakers, a full duplex echo cancellation unit, connections to other workstations, and a card that plays/records and provides a telephone interface. All of the audio devices are interconnected with an analog audio crossbar switch to allow one-to-many connections of audio signals. The current thrust of the VOX Audio Server development is for building research prototypes to demonstrate the utility of the server and, more generally, desktop audio applications. The status of this project is unclear, since the chief investigator no longer works at the Olivetti Research Center.

SEPARATING MULTIMEDIA HARDWARE FROM WORKSTATIONS

Most current generation multimedia applications assume a particular hardware configuration, and will only communicate with similar configurations. This approach breaks down in any reasonably sized organization where there is a heterogeneous mix of workstations.

One solution to this dilemma is to partition the software for the audio application into two separate processes - a server process controls the specialized media hardware, and a client process which issues requests to the server process. Once this software partition has been made, it is then possible to run the two processes on separate processors linked via a computer network. Thus, the workstation and the audio hardware can be physically isolated. With this configuration, one or more workstations can share expensive hardware. Also, various media hardware items can communicate directly and so place a minimal load on the controlling workstation.

Figure 1 illustrates a theoretical setup. With this simple setup, we can demonstrate a number of features of the system. First, we have two rooms, each containing a multimedia server with at least a microphone and a speaker (or headphones) attached. This allows users in both rooms to send and receive voice mail, and to have real-time conversations over the network (bandwidth permitting). Multimedia server A is attached to a telephone line, thus allowing the user at workstation 1 to have automatic telephone dialing facilities (from, say, an online Rolodex facility), to have the telephone answered automatically, and to have telephone messages digitized and stored for later playback. All of these features require additional hardware facilities in the multimedia server, but are feasible. In Room B, users of workstation 2 and PCs 1 and 2 have access to the same telephone facilities as the user in Room A; but, obviously, not at the same time. They could elect to use the FAX facilities of server A to send and receive FAXs. Using direct server-to-server communications, telephone calls received at server A can be redirected to server B if required.

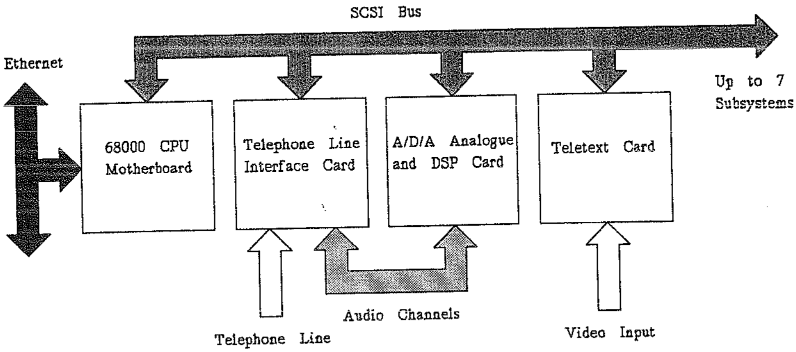

Figure 1: Typical network configuration.

THE PROPOSED MULTIMEDIA SERVER

We will begin by defining the types of applications which we envisage for the audio server, and then identify the characteristics that the server requires in order to support these types of applications. The order of the discussion of the applications does not in any way reflect their relative importance.

Applications

- Follow-me messaging. A number of computer networks have utilities which enable one to determine a particular user is on the network, and to locate where they are currently working. It could be expected that with this knowledge, users could elect to have a multimedia server receive the calls from their office telephones and to notify the user over the network. It is also not unreasonable to expect that the telephone conversation itself could take place over the network. The audio data is transferred between the multimedia server which receives the incoming telephone call and the nearest multimedia server to the user.
- Voice Mail. This feature is with us now. One can get this service on PC networks. However, we aim to provide a workstation independent platform on which to run this application. This means that users of vendor specific workstations, PCs, and X display terminals all have access to this facility. There will be no need to change the underlying mechanism, merely the vendor specific user interface to the network facilities.
- Conferencing. In a similar fashion, we can envisage conferencing taking place over the network, and even include users from the telephony network.
- Music composition and playback. It is planned that we provide a MIDI (MIDI [1985]) interface to enable communication with the wide range of MIDI compatible musical devices and keyboards.
- FAX communications. When used with FAX facilities, the server can forward FAXs to people (or to other computers), and receive FAXs for later distribution.
- Answering Machine. The multimedia server can act as a semi-intelligent answering machine. When combined with pre-recorded phrases or a speech synthesizer, it is possible to enter into

a restricted dialogue with a user. With the provision of speaker identification mechanisms, it is possible to enter into a variety of dialogues - one for the spouse, another for close friends, and an entirely different one for that persistent insurance salesperson.

Characteristics

To satisfy the needs of the abovementioned applications, the server architecture must demonstrate the following characteristics:

- Real-time behaviour: We need to be able to respond to events with predictable response times. This will necessitate some form of queuing mechanism similar to (Arons [1990]) that minimizes the overhead of processing audio requests and events at time-critical moments.
- Device independence: The client applications need to be shielded from the idiosyncrasies of particular audio components.
- Extensibility: The architecture should allow new types of audio (and video) devices to be simply integrated into the server, and be able to support new uses for these services through a hardware independent interface to the operating system.
- Dynamic reconfiguration: It must be possible to dynamically reconfigure the audio connections. In the case of a telephone answering application we need to be able to rapidly move from speaker recognition mode to recoding mode to playback mode, often several times, during the one conversation.
- Multitasking: By its very nature, a multimedia server provides a number of services, from low level accessing of the network, through processing of network messages, to controlling multimedia service cards. In order to allow the use of various services concurrently, it is necessary that we have an operating system capable of supporting multitasking.

Basic Architecture

Figure 2 details the subsystems of the multimedia server. The server is broken down into a motherboard which provides the basic network interface and coordinates communication with the specialized peripheral cards.

Because of the complexity of the tasks performed by each of the subsystems, each card is provided with its own microprocessor. This helps to reduce the load imposed on the motherboard, and allows us to use the relatively low speed and low cost 68000 CPU for nothing more than network control.

These subsystem cards provide the following capabilities:

- Telephone network interface. This card provides the standard line interface, with pulse or DTMF dialing capability, on/off-hook detection, ring detection, etc. The output from (and into) the interface is the telephone audio signal.
- 12 bit A/D and D/A and DSP interface. This provides 4 channels of high quality audio (2 input channels and 2 output channels), with user selectable sampling rates up to 44.1 KHz. We require one high performance DSP processor (at least) to enable us to perform speech synthesis, speaker verification, speaker identification, and speech recognition. There are a number of projects within the department (mainly in the neural net area) which will contribute software for these applications.

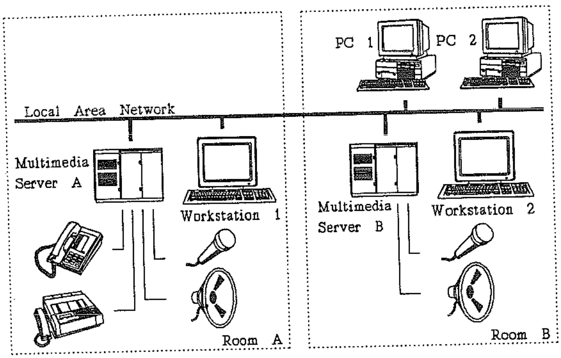

Figure 2: Multimedia server configuration.

- Audio crossbar switch. This is to allow audio signals to be routed between the various cards that are available.
- Teletext card. Users can access any page of Teletext over the network and, with appropriate applications software, maintain continuous connection with a page to monitor events (e.g., such as storm progress).
- Weather FAX card. Similarly, we can provide FAX transmissions from the weather bureau in a form suitable for display on a wide range of workstations.
- ISDN card. When we are connected to an ISDN exchange, we will complete this card to enable connection to a wide range of services being developed for ISDN.

The motherboard contains the network interface, a CPU (Motorola 68000), 1-4 Mbyte of DRAM, 2 serial ports, a parallel port, and a Small Computer System Interface (SCSI) bus and its associated controller. Specialized functions are provided by the subsystem cards which communicate over the SCSI bus. Because the transfer rate of the SCSI bus is approximately 1.5 Mbytes/s, we limit the number of subsystem cards which we connect to any one audio server. This restriction will be eased when we move to more exotic technologies later in the life of the project. For now, if more subsystem cards are required, then we simply add another audio server to the network and place the additional cards in this server. A database used by application software can be updated to reflect the presence of the additional server and its associated services.

PRESENT STATUS

The motherboard is complete and operational. The XINU operating system (Comer [1984]) will be ported to the motherboard over the coming Christmas vacation. XINU provides both the multitasking capability as well as the network code for the network protocols we are using.

The first of the subsystem cards is now completed - the Teletext card. The next two cards scheduled to be completed are the telephone line interface and the A/D/A cards.

REFERENCES

Ambron, S. and Hooper, K. [1990], *Learning with INteractive Multimedia: Developing and Using multimedia tools in Education*, Microsoft Press, Redmond, Washington.

Arons, B. [January 1990], *VOX Audio Server*, Olivetti Research California, Technical Report.

Arons, B., Binding, C., Lantz, K. A. and Schmandt, C. [April, 1989], *The VOX Audio Server*, 2nd IEEE International Workshop on Multimedia Communications.

Comer, D. [1984], *Operating System Design: The XINU Approach*, Prentice-Hall, Englewood Cliffs, New Jersey.

MIDI [1985], *MIDI 1.0 Specification*, The International MIDI Association, Technical Report, 11857 Hartsook St., North Hollywood, CA 91607 USA..

Schmandt, C. and Arons, B. [August, 1984], *A Conversational Telephone Messaging System*, IEEE Transactions on Consumer Electronics CE-30, xxi–xxiv.

Teitelman, W. [1984], *A Tour Through Cedar*, IEEE Software.

Terry, D. and Swinehart, D. [November 1987], *:Managing stored voice in the Etherphone system*, 11th ACM Symposium on Operating System Principles.

Zellweger, P., Terry, D. and Swinehart, D. [March 1988], *An Overview of the Etherphone System and its Applications*, Proceedings of the 2nd IEEE Conference on Computer Workstations.