Panel Session

Speech and Audio in Window Systems: When Will They Happen?

Co-Chairs: Barry Arons, *Olivetti Research Center*
Chris Schmandt, *MIT Media Lab*

Speakers: Michael Hawley, *NeXT Inc.*
Lester Ludwig, *Bellcore*
Polle Zellweger, *Xerox PARC*

Good afternoon. Boy, I can't see anything out there. I assume you all can see me -- thats why these lights are here. My name is Chris Schmandt from the Media Lab at MIT. I'm co-chairing this panel with Barry Arons, who is sitting over here. It's actually quite a pleasure to co-chair this panel with Barry. We've been working together off and on for more years than I care to remember.

This panel has a long ridiculous name. Basically it's about audio and window systems and workstations. I'm wearing two hats here. I'm going to spend a minute or two introducing the panel and then I'm going to spend some time talking about my own segment of the panel.

We're going to try to be a panel as opposed to a series of five mini-papers that never get published. In other words, we're going to try to keep our presentations relatively short, then segue into a series of prepared questions that the panelists are going to answer amongst themselves. Then we'll open the floor up for questions.

In some ways this is a very incestuous crew. We've all known each other for quite a while. We have different slants and we're actually going to try to focus on those slants a little bit. So if we disagree with each other, that doesn't necessarily mean we really hate each other. We're all friends.

Where this panel is coming from is a surge of interest in audio, and multimedia, in general, in computer workstations. The Macintosh has had audio for quite a while -- you may or may not choose to call that a workstation. The NeXT computer sort of surprised people by having fairly powerful DSP and audio in and out. You'll get a demo of that later if you haven't seen it. The Sun SPARCStation has come out with some primitive digital record and playback capabilities.

On the other hand, there's been interest in voice in computer workstations for years and years, and what we've seen so far is that voice really hasn't had very much success. There have been a number of products that have come and gone. What has become popular has been centralized service -- specifically voice mail. Voice mail is tied in more to a PBX -- and the interface is more like a telephone than it is a mouse and window system, in the computer workstation interface.

Obviously, window systems are here to stay. We're not suggesting that audio is going to replace the graphical paradigm, but rather have to interact with it.

On the other hand, everybody has a telephone. People had telephones on their desks before they had workstations, and we talk all the time at work. Voice really is a fundamental component of the way we talk, the way we interact with each other.

What we're seeing in terms of the technologies showing up in these workstations is higher bit rate coding. Gone are the days of unintelligible low bit rate linear predictive coding or something like that -- except for specialized applications.

Speech recognition is here, but it's in its infancy. Text-to-speech -- it's around, it's difficult to understand. You can learn to understand it.

Telephony is obviously part of this set-up if we're dealing with audio. We don't know whether it's going to be analogue or digital. Is it going to be plain old telephone or is it going to be ISDN?

Those are some of the issues that we're going to be talking about in this session. As I say, we're going to try to keep each of the speakers to a relatively short period -- and now I can put on my other hat. (puts toy plastic headset on -- laughter)

Some people ask me whether speech recognition is a toy or not. Yes, it is. It's sort of a fun toy. Speech technologies are in general fun. I was originally hoping to be able to play this out to the audience. But I don't think it's going to work well enough. This is actually a kid's toy -- $50 at Toys R Us. Speaker Independent Isolated Word Speech Recognizer -- "yes", "no", "true", and "false". It will take you on tours about dinosaurs and things like that.

From my point of view, the key for what we can do with voice has to do with understanding its advantages and disadvantages and the comcomitant user interface requirements leading us to design reasonable applications for it.

Voice has some advantages. It's very useful when your hands and eyes are busy; you're looking at a screen, you have your fingers on the mouse. Sometimes it's intuitive; we learn to talk at a very early age. People talk to their computers even if the computers don't have speech recognition. (laughter) Usually it's expletives -- especially with UNIX. (laughter) Voice really dominates human-to-human communication. No matter what we're doing with E-Mail and FAX, the bottom line is we just still have to spend a certain amount of time physically speaking to each other.

Telephones are everywhere. If I can turn an ordinary pay phone into a computer terminal, suddenly I have access from all over the place.

From my own work, this suggests a heavy focus on telecommunications. The kinds of systems that I'm building are really designed to use voice in a communications kind of environment. On the other hand, there's many, many disadvantages of voice. It's very slow. 200 words per minute, 150-250 words per minute. That's less than a 300 baud modem and who uses those any more.

Speech is serial. You have to listen to things in sequence. It's a time varying signal by definition. And it requires attention. You have to listen to what's going on, as opposed to simply scrolling it by and stopping it occasionally.

My way of characterizing this is to say that speech is "bulky". Yes, it takes up space on the file system, but most importantly you can't "grep" it, you can't do keyword searches on it. It's hard to file, it's just hard to get any kind of handle on it. It takes time.

Finally, speech broadcasts. If my workstation is talking to me and you're sitting in my office, you're going to hear what it says, which is very different from if it appears as text. In fact, if it appears as text, and I'm sitting in front of the screen with these kinds of tiny bit map fonts that we tend to use, I'm probably not even going to be able to read it -- much less you.

This has some user interface implications. One is that it suggests that we would like, where possible, to have graphical access to sounds. I'm going to show a video in just a second, showing you an interface to audio built under the X Window System, designed to give you some kind of a graphical context, so you can mouse around and perhaps use some visual cues to keep track of where you are in the sound. If you could roll the first piece of one-inch, please.

This is a sound widget.

— VIDEO TAPE BEING PLAYED —

Thanks to Mark Ackerman for doing the coding. As you can see, this is integrated in the rest of the window system and this is a recurring theme, and what I'm talking about is the need for *integration* -- and we'll get back to that in just a second.

Another issue here with voice is because it's so slow, you have to support interruption. In the case that I just showed you, while playing a sound, you could mouse around and stop it and play any other sound. You could drag the little cursor around and let go and hear a piece over. If you're calling up from a telephone, you always have to be listening for the user to respond to a touch tone while you're playing something. You don't want to leave somebody where they have to listen to a three-minute message in order to do the next thing. Predictably what they'll do is hang up and walk away. That's not a useful user interface.

Finally, because of the difficulties of speech, this suggests it might be useful as an auxiliary channel. We're not trying to replace the keyboard. We're not trying to replace the mouse -- or maybe we are trying to replace the mouse. But what I'm suggesting is that for some of the kinds of operations that I'd like to do on a computer, voice is a side channel which may convey another channel -- another domain of information. And obviously, in terms of auxiliary channels, if I'm not in a situation in which I have a terminal, I have a keyboard, and a CRT, and a mouse, then the world is wide open for voice, calling in from a telephone. You can only do so much with a 12-button touch tone keypad.

From my point of view, and the work that we're doing currently at the Media Lab, the key really is integration. Sound is not a medium that exists without respect to the other kinds of things that are going on in the workstation. As I already said, there's a need for graphical representation. So here right away we're tying in graphical user interface and the audio as data.

What we would like is for voice to become an integrated part of whatever window system we're using -- because obviously, window systems are simply part and parcel of our computing environments these days. We'd also like speech to become a ubiquitous data type -- something that we can use in a variety of applications -- in just the same way that we use text. We don't really think of text as a data type. Text is just the medium that we use to interact with things. Sometimes it's a command channel when I'm typing into the shell. Sometimes it's data when I'm editing a document, and we'd like voice to be able to do the same thing.

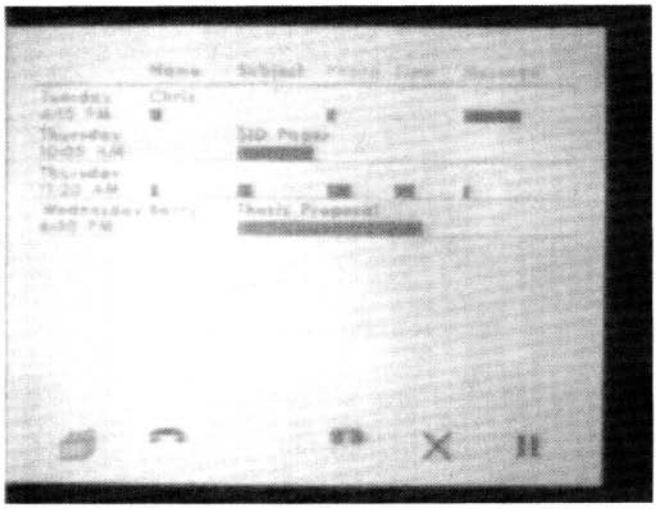
This really suggests that we're looking at multiple applications. In the rest of my talk I'll describe some of these -- a number of different voice applications. And those applications are going to cover a range of functions -- things like editing and document creation and messaging. And they're going to coexist with the kinds of things we already do on our workstation. We're not going to stop using our workstations for editing programs because all of a sudden we're going to start recording voice on them. This is going to suggest that those applications are going to have to have access to resources. Audio resources are going to be shared just like bits on a screen or shared by a window system. Barry is going to talk about this in some detail in his talk.

Finally, in terms of a window system, I believe that we're going to need a technique for doing multimedia selections. One of the things that window systems get us is the ability, under user control, to select a piece of data and stick it in another application -- call it selection, clipboard, cut and paste or whatever. Things start to get interesting when the object of the selection -- it may be text, or it may be audio, or it may be a sound file; it may be a sound file segment.

In fact, from the point of view of my interest in communication, the selection might be a telephone number which is temporarily represented as text, or it might be an address, or it might be a reply to a telephone message. We need to be able to support the ability to shift these different kinds of data and different messages associated with that data back and forth between applications.

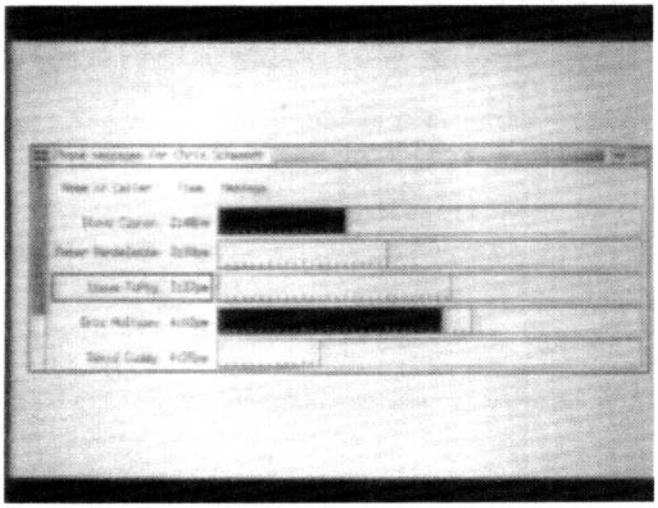
So basically, the long and the short of it is that I'm saying that speech *really* needs to become integrated into the window system in many, many different ways, in order for it to achieve the kind of functionality that we currently have with text. It's never really going to get there, but it may be able to approach it asymptotically.

In terms of applications, answering machines and voice mail, they're almost the same. Voice mail gives you the ability to forward messages. Here is an old slide from an old system called Phone Slave, that Barry and I built a long time ago at the Media Lab with graphical representations of sounds.



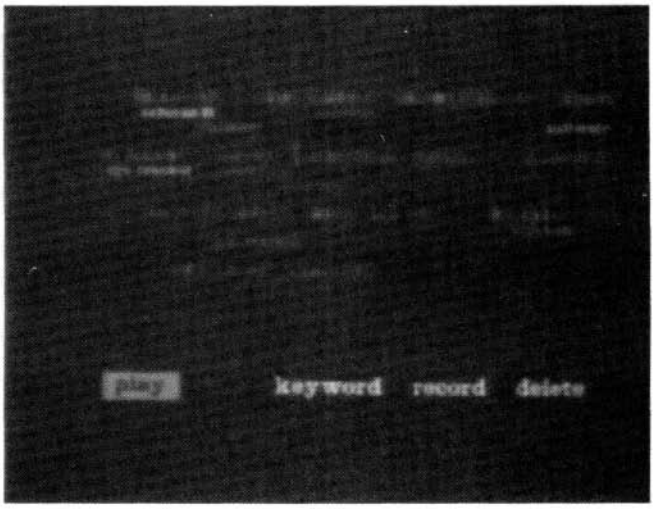
— SCHMANDT - SLIDE 10 —
© Chris Schmandt

Here is a similar one that we did just the other day that I showed you on tape -- the X interface.

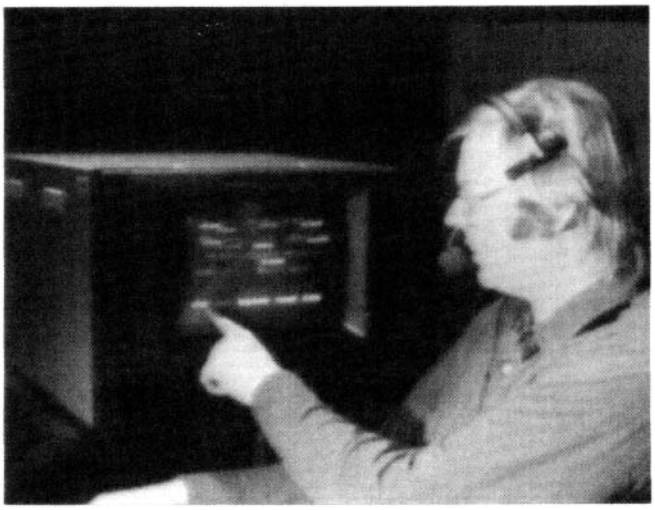


— SCHMANDT - SLIDE 9 —
© Chris Schmandt

We need to be able to support annotations in multimedia documents. I'm not going to say much more about that because Polle has plenty to say on that. We need to be able to edit audio. Here is a very primitive audio editor that we did a long time ago, using a touch screen called the Intelligent Ear.



— SCHMANDT - SLIDE 11 —
© Chris Schmandt

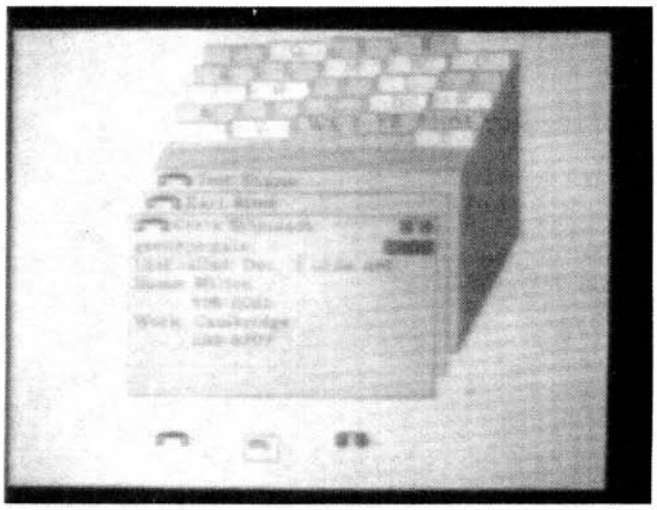


— SCHMANDT - SLIDE 12 —
© Chris Schmandt

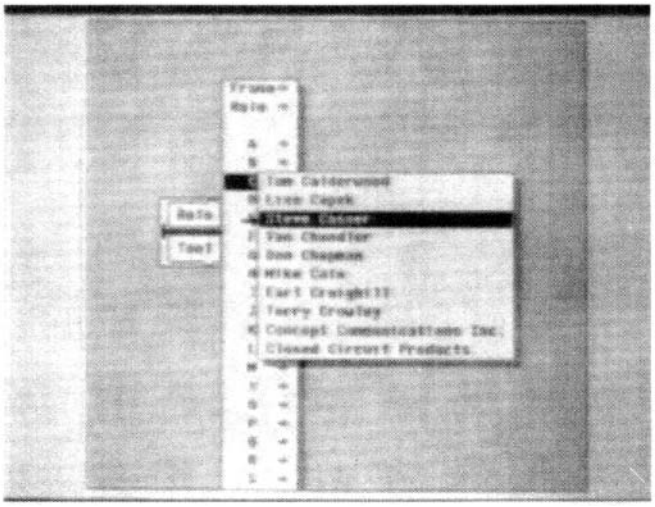
It shows average magnitude function as a way of describing the contents of a sound graphically. Here is S-Edit, an audio editor based on the X Window System that Barry and I did at Olivetti not too long ago.

It gives you periods of speech and silence and you can manipulate them.

You also are going to need -- in terms of communication -- tools for doing speed dialing. Here is a Rolodex shown in two forms.

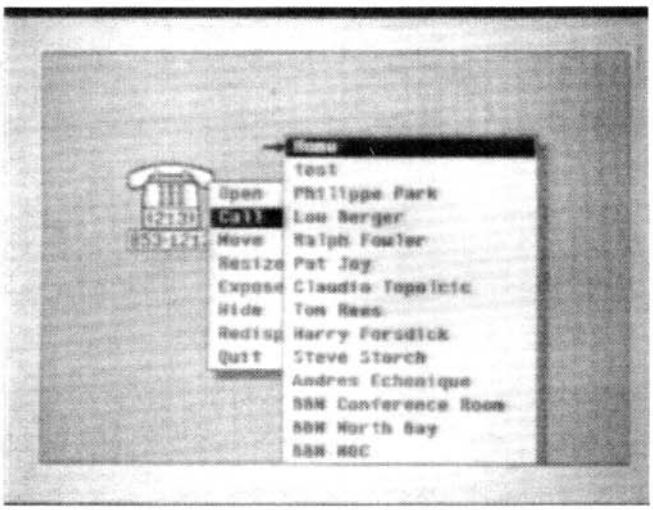


— SCHMANDT - SLIDE 13 —
© Chris Schmandt



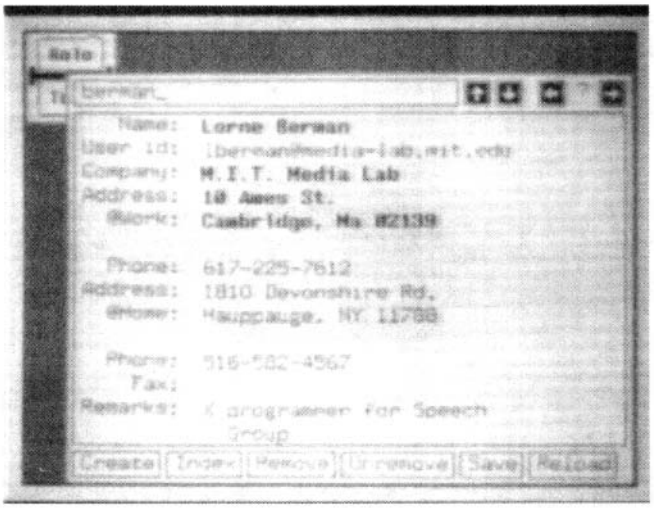
— SCHMANDT - SLIDE 15 —
© Chris Schmandt

Speed dialer -- this is actually running under Sunview.



— SCHMANDT - SLIDE 16 —
© Chris Schmandt

A Rolodex under Sunview, and you pop up a card; here's the X version of it.



— SCHMANDT - SLIDE 17 —
© Chris Schmandt

And a variety of these tools interacting with each other.

One of the other things that we can buy is remote access. I've got another piece of tape that I'd like to show, showing you calling in from a remote location and reading your electronic mail and your voice messages. This is a piece of the Phone Slave that Barry and I did so many years ago at the Media Lab. If we could have the three-quarter inch, please.

— VIDEO TAPE BEING PLAYED —

Basically, I was getting text messages synthesized to me. I was hearing voice messages. One other final application is using voice to navigate *around* a window system. In this sense I actually am talking about replacing the mouse. If you could show the last piece of one-inch video. This is a real short

segment here. Windows have names. I speak a windows name and I jump to it. Could we have the one-inch, please?

— VIDEO TAPE BEING PLAYED —

That's basically all I have time to say. I'd like to introduce the next speaker, which is Lester Ludwig from Bellcore, who is going to talk about a variety of uses of audio, including some non-speech audio, which is not the place that I've been coming from.

Lester Ludwig
Bellcore

I can't see any of you, but I guess that comes with the territory in a live televised presentation. My main role in this panel, I was told, is to function as the blooper since this Session competes with the SIGGRAPH video blooper theater. As Chris is saying, our work deals with all forms of audio, not just speech.

I founded and am affiliated with Bellcore's Integrated Media Architecture Laboratory which maybe many of you people in the graphics community haven't heard about. I guess at some point maybe we ought to try to put some full papers into SIGGRAPH. People in the Internet community are probably fairly familiar with it because of some of our activities with various federal agencies. Many multimedia folks in the PC and workstation industry are also familiar with our work. Since this is probably most folk's first exposure to our project, I'm going to dwell a bit on what we're doing in the general area of multimedia computing and its communications needs.

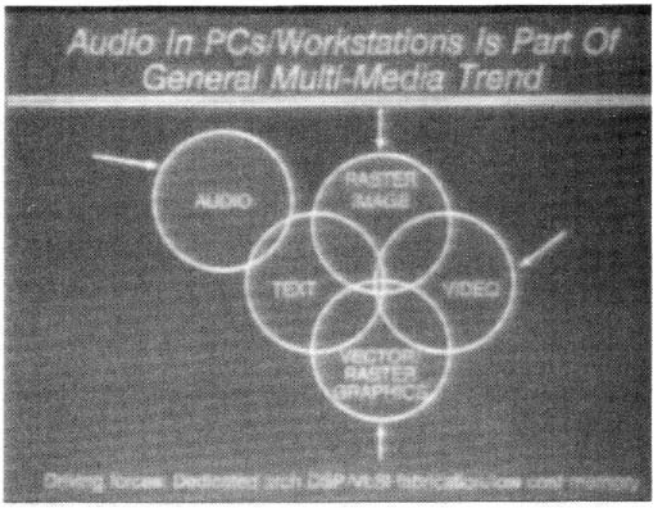
I'll also gain by saying a little bit about the general work we're doing because it motivates the audio windowing system that I'm here to talk about. Later there will be another discussion about the VOX system, which uses a different set of visual windowing ideas, but more from the (audio) resource management side. The audio windowing I'll be talking about is more in analogy with the user or presentation aspect of a windowing system.

So -- I'll begin formally with a summary of what I'm going to say. I'll start with identifying the general roles of audio within multimedia, and maybe how audio obtained the role it now has. Then I will talk a little bit about our Bellcore multimedia prototyping so you know the framework of where we're coming from, and I'll also probably throw in some plugs about the importance of your world (the computer community) and my world (the telecommunications community) working together which is long overdue and something all of us need.

I'm going to then focus in on the main part of my talk, which is this notion of audio windowing; I'll say a little bit about why it's needed and how it's done and what lies ahead with it. I hope you'll enjoy it all.

So I'll put on the next slide and start out with how audio in PCs and workstations is part of a larger general multimedia trend. I guess one point that I wish this notion of multimedia and hyper-media -- especially hypermedia -- would sort of disappear. In a way it really is all information, and with the proper data structures, hardware, and operating systems individual media details will become increasingly less important.

In the past there have been differences in media that have been largely because of technological artifacts and how technology first started tackling its handling of these kinds of information. So first I show this picture, a little bit like pictures were used to sell the Media Lab when they were talking about the publishing industry and the computer industry and video all merging together.



— LUDWIG - SLIDE 4 —
© Bellcore

The idea is that in the past there's been separate domains of text and video and raster graphics. First raster images, raster graphics, and vector graphics started merging and then they started bringing text in because you could blit text from cached font libraries... later you could do text with splines and stuff and then video started to get pulled in because it was a raster. So you had the workstation industry sort of bringing that gang of four together there. Also a lot of computer audio hardware starting to show up -- speech synthesis, speech recognition, which turns audio in and out of ASCII, and then audio was being stored in byte format along with other data.

Audio is just part of this -- and that's good because we would sort of like audio to be just like any other sort of media type. When you need it and when it's comfortable, it should be available and supported in in some sort of natural way. I guess that's a goal a lot of us have and a lot of us would like to see happening. And we have to do it a little at a time.

Let me say a little bit about our work at Bellcore. We've been involved with multimedia for quite a while. I came there in August of '86 and started this project called the Integrated Media Architecture Laboratory. The first thing we did was incorporate a Parallax board, which some of you have probably heard about. It digitizes video and puts live video on the screen and allows you to do graphics operations on the video and pointer overlays and so forth. We had that for a while and we're doing some multimedia networking stuff and after we fooled around with it for about a month or so we realized we needed a windowing system. So we decided to go for the X windowing system, and pretty much as a direct function of that need, Bob Goodwin from Parallax came out about New Years' Eve and worked on it for us for a while. He did all the work; I watched. He ended up making the beginnings of what was the X server and the driver for the Parallax board that still the only commercially available way I know of to do live video under X windows. There are many other projects just around the corner, but Bob Goodwin is really sort of the unsung hero of all this -- not only on the software side, but he also designed all the hardware that does this stuff, in the gate array domain, which is quite difficult at the speeds that these things have to be done at,

all of this in his living room with his own software design system; he's a pretty admirable guy.

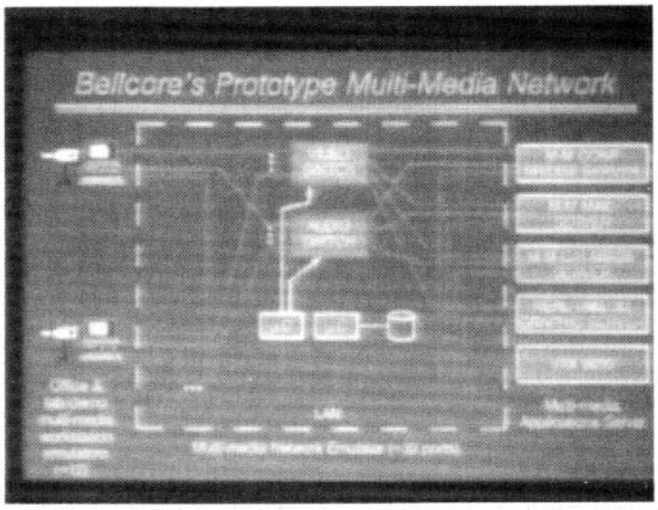
Anyway, the guy sitting at the terminal in the photo -- he looks a lot better than I do actually -- I'm losing my hair -- is Mark Levine, who later helped legitimized the X server for the Parallax board.

— LUDWIG - SLIDE 3 —
© Bellcore

At that time he was with MIT Athena and he made the for-real X server. From there things go on and on. Martin Levy was also involved with all this.

Anyway, so that's just part of the terminal. I did want to point out that Bellcore did pioneer the first work to put video under X and also that we initiated the X video extension standard, which Todd Brunhoff from Tektronix has really done all the leg work and gotten all the things that happen.

This is our network, and the intent is to study multimedia communications.



— LUDWIG - SLIDE 6 —
© Bellcore

We've got terminals like in the photo and we have our multimedia network emulator that's a stand-in for something called Broadband ISDN and a number of other networks that may come in the future.

The key thing is that there is some terminals on one side and a collection of multimedia servers on the other side. There are conference bridges whose use was shown in the previous slide.

There was a multimedia conference going on in that photo. The video window is broken into four parts and there's a five site conference going on there. Other windows can be shared using SharedX which was talked about at the multimedia session yesterday.

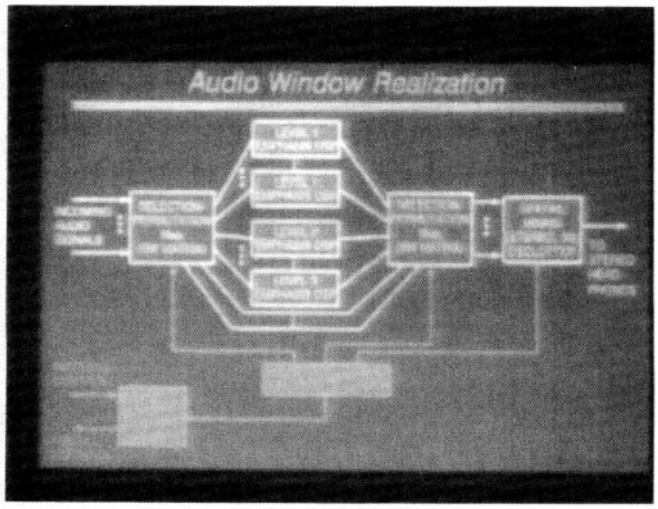
Anyway, multimedia conferencing is one of several things offered by the network. There's also a multimedia mail server. There are multimedia data base systems, and some real-time interactive 3D graphics. The main reason why we're doing this is because, (as Dave Clark points out,) information and people are not always in the same place at the same time and we really need to get these things to be more in recognition of that. All this desktop multimedia world is fine, but as soon as you have it, it's going to be just like the current needs to network desktop computers and workstations.

So now let me just pull this all back together to the audio theme. So if you look at all those applications that we have built, there is a lot of audio. In fact there's different types of uses for audio. People are using audio for some pretty novel things. There is a speech which I guess is most of this forum is talking about today. There's also a music and environmental stuff like recordings you may make of nature, or the surroundings, or some sort of machine that you're studying, or combinations of these things. They need more general audio channels and appropriate ways of managing all the audio from the user's viewpoint.

The audio windowing system that we're trying to put together is intended to be something much like we did for the visual stuff where we tried to bring all visual media together in a single user interface system. We'd like to see the same thing kind of done for audio, multiple audio sources put together in a similar "feel" and functionality windowing system of some sort.

So, how do you do that? Well, you depend a lot on what is known about psychoacoustics and electronic music and different things. What we ended up doing was putting a system together using electronic music systems to do hierarchies and some systems that do spatial audio imaging. One main idea is the display of several simultaneous sources at the same time. Another main idea is we use is a notion of hierarchy. Most of the other work that you'll hear about today deals with turning one source off while turning another source on, or maybe using a mixer. Our system is a higher level of sophistication. In particular we use a notion of a window manager for the management of multiple sources using spatial and rank metaphors.

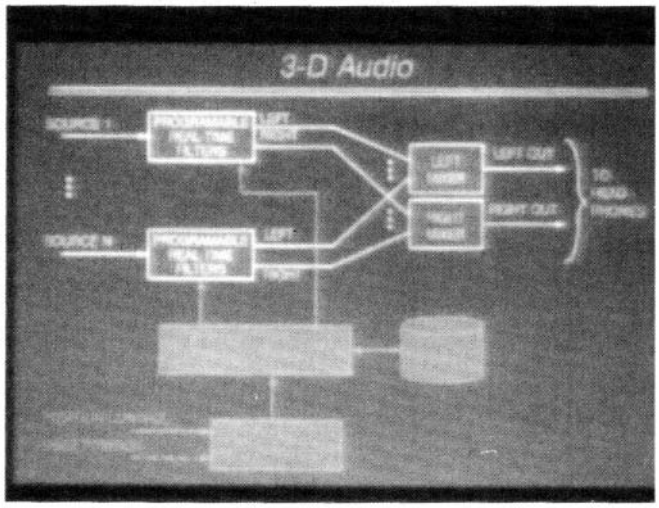
Here is a basic way of how you'd implement something like this.



— LUDWIG - SLIDE 10 —

Start with a group of incoming audio signals and first choose which of them you want to display. Then you have a signal processing stage that can do emphasis and that is not turning volume up and down; it's using some psycho-acoustic processing. If Todd Rundgren was at SIGGRAPH this year, he'd know exactly what I was talking about. But I think many of you who are involved with audio probably know about aural exciter and psychoacoustic imager systems.

There's another selection permutation map that's there. Then some spatial mixing which in the simple form is stereo audio, just panning between left and right. But it can be a little bit more sophisticated. In particular, we're using Scott Foster's 3D audio system, which is the same thing that Scott Fisher is using in his NASA Ames research with the artificial environments.



— LUDWIG - SLIDE 9 —

There are basically some programmable real-time digital filters. Coefficients are selected from a coefficient library and they synthesize what your ears would hear if one sound source was there and one sound source was somewhere else. What we also hope to be able to do is have position control come in and you could even have user feedback by putting a Polhemus sensor on your headphones and as you move your head around, the sound imaging compensates.

The last slide is our studies in progress. We're of course trying one focus on teleconferencing. I mean, it would be nice to have these sort of spatial metaphors to try to help you organize what person is in what place -- maybe associated with a visual image. But there's another issue about very large conferences where you might need to bring people in and out of focus groups and then having some sort of on the side outside the periphery of what you can see and then maybe back behind your head where the visual stuff doesn't work too well... those can be some useful things.

Also we're interested in trying to work within ISDN bandwidths, which is more like the seven kilohertz rather than the high fidelity channels we're working on. We're doing some pretty cool stuff too using the Data Glove with the 3D information management, and we're trying to combine that with this 3D windowing system we're trying to build. Sort of like in the spatial data management days -- you can move through and affect these information spaces. Using binocular vision and the data glove. We have a couple of summer students that are doing most of the work on this. Natalio Pincever, who will be working with Chris, and Michael Cohen who is from Northwestern, who have been doing some work on that.

So that's all I've got to say, and I'll sit down and turn the proceedings back over to the presider here.

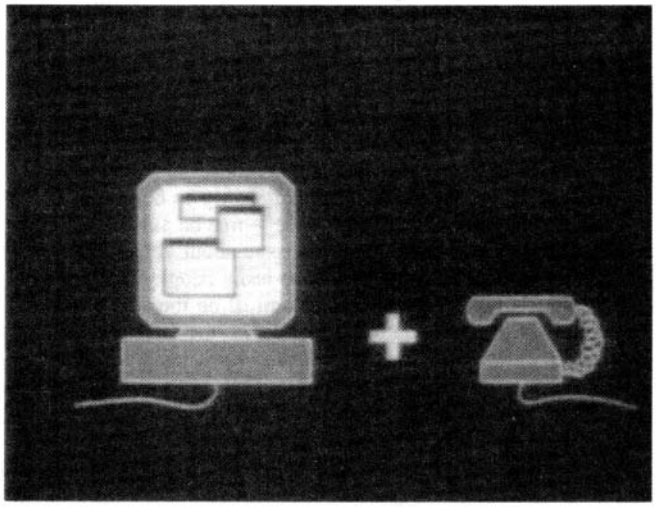
Moderator
Chris Schmandt
MIT Media Lab

Thanks, Lester. The next speaker is Polle Zellweger from Xerox Palo Alto Research Center.

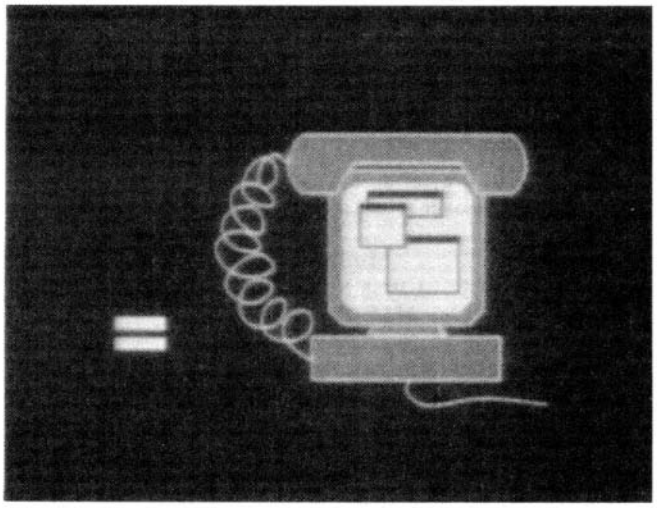
Polle Zellweger
Xerox Palo Alto Research Center

I'm going to talk about some of the work that's been done over the past five years at Xerox PARC in the Etherphone project.

The goals of this project have been to combine the functions of the telephone and the workstation to create a multimedia workstation



— ZELLWEGER - SLIDE 8 —
© Xerox PARC



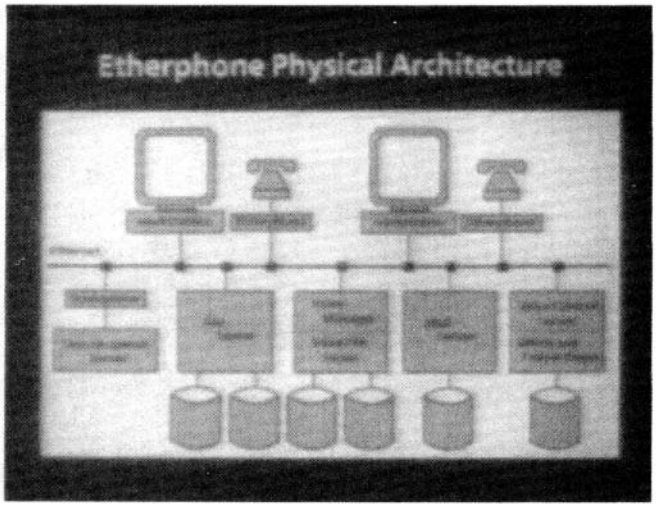
— ZELLWEGER - SLIDE 2 —
© Xerox PARC

that can place, receive and manage calls better than a conventional telephone, and can use recorded or generated voice throughout the workstation environment -- as the previous panel members have been discussing -- much as most systems use text and graphics today.

We want documents, programs, and user interfaces to have access to voice and telephony. Along the way we've had to pay considerable attention to constructing a voice system architecture that would provide these capabilities uniformly in our distributed workstation environment.

Now since Chris has already discussed telephone applications a great deal, I'd like to concentrate on recorded voice. What I'm going to do is I'm going to show you a vertical slice through the Etherphone system that describes how we implement and use voice recording and editing.

Our system is based on Etherphones,



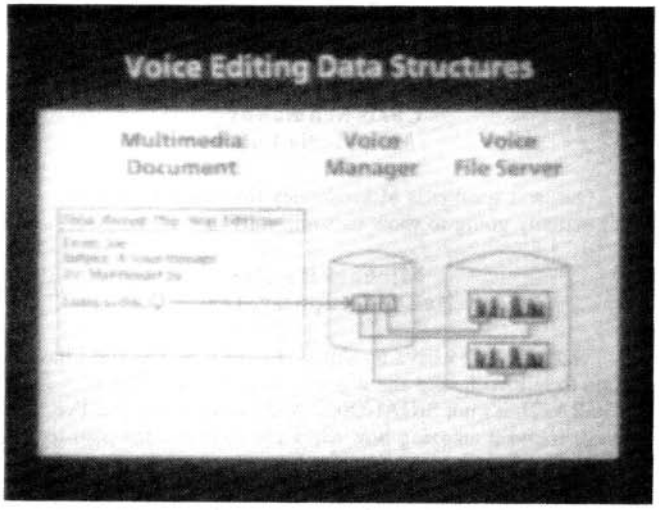
— ZELLWEGER - SLIDE 1 —
© Xerox PARC

which are custom microprocessor-based telephones that transmit telephone quality digital voice over the Ethernet. To maintain security of an individual's telephone conversations, all voice traffic is encrypted using DES encryption.

The voice control server controls the operation of all of the Etherphones by remote procedure call over the Ethernet. We also have a central voice file server and a text-to-speech server. There are about 50 Etherphones in daily use at PARC.

We had several goals for handling recorded voice in our distributed environment. First of all, we wanted it to be sharable among a variety of different workstations. We also wanted it to be editable and we wanted it to be available to many different workstation applications, for example, document editors, calendar programs, or whatever. However, as Chris has already said, voice is bulky. In our environment we use 64 kilobits per second voice, and that gets large pretty quickly.

The slide on your right shows a diagram describing voice editing.



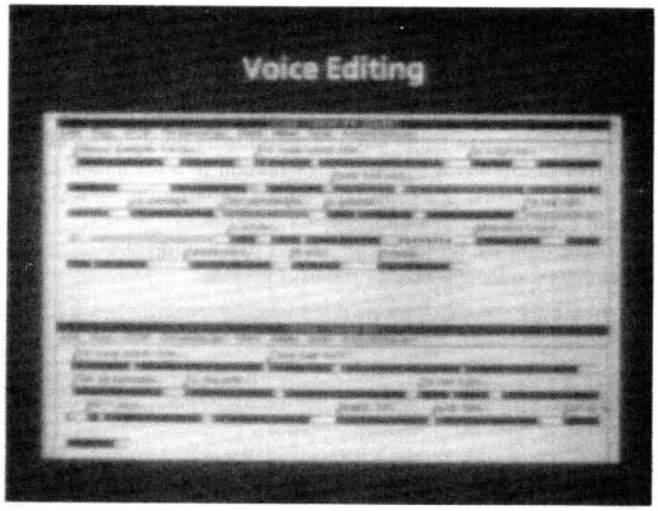
— ZELLWEGER - SLIDE 4 —
© Xerox PARC

This multimedia message here -- the mail message -- is on the user's workstation, and the other two elements -- the voice manager and the voice file server -- are central server applications. They don't live on the user's workstation.

We record voice by setting up a conversation between a user's Etherphone and the voice file server. That stores encrypted voice samples on the voice file server. The voice management server then provides editing operations analogous to string operations, such as replace, concatenate, substring, and so on. These operations create immutable voice descriptors in the voice manager data base. They do not copy or decrypt the voice.

Applications on the workstation use voice only by reference. Because edits always create new voice descriptors rather than changing old ones, we can store the voice centrally and still share it among a variety of different workstations. Furthermore, since applications use voice by reference, existing applications don't have to be modified in order to handle embedded voice. Finally, we use a modified version of reference counting to provide garbage collection of unused voice descriptors and voice samples.

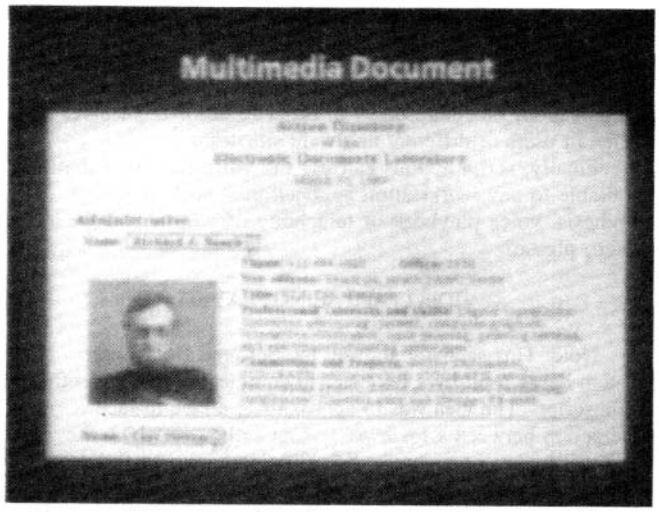
On your left is a view of the user interface to our voice editor.



— ZELLWEGER - SLIDE 12 —
© Xerox PARC

It shows a sound and silence representation of the voice. Sound is dark and silence is light. Users can cut and paste voice recorded from different sources using exactly the same keystrokes that they would use in our text editor. Or they can record new voice at any point, as shown here in the black triangles. Color provides a brief editing history to augment the sound and silence profile. Users can also annotate the voice with text.

On your right is a multimedia document.



— ZELLWEGER - SLIDE 3 —
© Xerox PARC

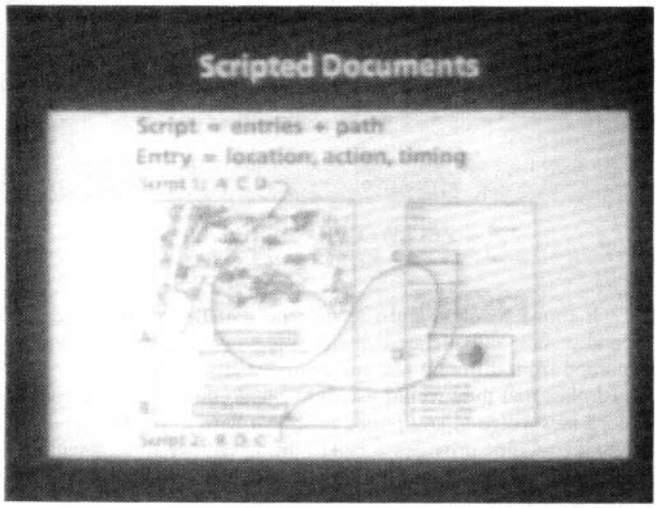
This little voice balloon here indicates the presence of a voice annotation. Voice annotations can be added to any character throughout the document and they don't modify the document's layout.

Now when we started to actually use voice annotation after we had implemented all of this, we discovered a new need. A document can accumulate many different voice annotations on many different subjects throughout its lifetime, and we needed a way to collect these annotations -- the ones on the same

subjects -- together and also order them, because the linear order of the document might not be the proper order in which to listen to the annotations.

As a generalized solution to this problem, we built the Scripted Documents hypermedia system. It allows users to easily create active paths through multimedia documents. A script is essentially a dynamic presentation of a document or a set of documents. It's especially suited to narration, as we'll see in a video clip in a moment.

The diagram on your right shows two multimedia documents with two scripts traversing them.



— ZELLWEGER - SLIDE 6 —
© Xerox PARC

Scripts can share entries and the underlying documents can be edited without disturbing those scripts. Scripts can also contain loops or conditionals in order to tailor themselves to different users or different hardware situations.

Finally, script actions can use the voice functions that are available to any workstation application -- such as speech synthesis, voice playback or telephony. Can you roll the video, please?

— VIDEO TAPE BEING PLAYED —

Note: This segment showed the Scripted Documents system being used by a reviewer to comment upon an electronic manuscript. The idea was to approximate a face-to-face interaction between a reviewer and an author, in which the reviewer makes comments while flipping back and forth through the manuscript and other documents to substantiate those comments. The reviewer selected areas in the text, added voice annotations, and created an ordered path (unrelated to the linear order of the manuscript) that included annotated areas in the manuscript as well as in reference documents. The author played the resulting script back to hear reviewer's comments.

In conclusion, we've found that ubiquitous availability in control of live and recorded voice allows improved communication and hence improved productivity. Second, it's important to manage recorded voice carefully in a distributed environment to promote sharing and ensure security.

Finally, I believe that audio will revolutionize documents. We'll have narrated documents and documents with scores. The simultaneous use of visuals and audio enhance communication and retention. Thank you.

Moderator
Chris Schmandt
MIT Media Lab

Our next speaker is Mike Hawley from NeXT Computer. He's actually going to show us some stuff live on the NeXT.

Michael Hawley
NeXT Computer, Inc.

That's right -- I have no slides. I have a NeXT, and I have sound to talk with, and show you -- so listen up. This is SIGGRAPH; it's not SIGAUDIO. And I have to say that I've always found it amazing how little attention is often paid to sound. For all the computrons and smart equations that get spilled into rendering problems and dynamics problems, when you make a film like Luxo Junior and put the sound in it, there's a Foley artist there squeaking a Luxo lamp to get the springs to make the right sound. There is a problem here because sound is a very, very valuable and hopelessly under used communications channel at the moment. And I think there's a lot to do to fix that.

Let me -- before I speak a little bit here -- just make sure that the computer is working. I'm going to try and play a sound and ... cross your fingers.... (Theme from *Superman*...) We actually had problems with this SCSI disk earlier, but there seems to be sound now.

The question that I'm most concerned with is *when* -- and not speech, but audio. I care a lot about general purpose audio. I don't believe that audio is just for speech any more, and I think it's high time that we pushed workstations into the great age of talkies. All of us now use silent computers and that's deplorable.

The question is how to change that. I looked back over some history that I found kind of interesting and two points stood out for me. Jack Foley and Vitaphone. Let's just talk for about two minutes about how sound was invented and how long it took.

Jack Foley was a sound effects guy at Warner Brothers in the 1930s, and he's responsible for "Foley effects." Those are, by and large, human nonvocalic sound effects, like footsteps and burps and nose crunches, all the Three Stooges noises, and Luxo lamp springs. Those are all Foley effects. The way they're done in Hollywood these days is with a Foley artist. You put a lot of rubble on the stage, watch the movie as it goes by, and "perform" the sound effects. Yet that whole part of the industry has bloomed into something that -- well, Ben Byrd at Lucasfilm managed to elevate into a whole field called "Sound Design," when he did Star Wars.

The kind of craftsmanship that goes into making a movie present itself acoustically in an appealing and compelling way, is something that I think computer scientists really ought to look a lot at -- especially as we begin to design computers that have sound as integral parts of their interfaces.

The other historical point that for me was a little bit more interesting was the whole advent of "talkies" and where they came from. From about 1900 to 1930 or so, people tried very hard to invent sound. There was a lot of skirmishing in the industry. Everyone knew they wanted to integrate audio into their presentations somehow, but they all had a different tack.

Edison invented Kinetaphone in 1895 and it was a flop. A very much McLuhanesque kind of progression. Not unlike combining a computer with a telephone, Edison took a giant photograph player and hooked it up to a projector. The phonograph was down by the screen and there were a system of belts of pulleys to try and keep it in sync. This is not unlike some approaches that are being taken right now by people in the computer industry to integrate audio into their workspaces.

There were a host of almost Balkan approaches to the problem. If you look back over the inventions, you find Cameraphone, Kinetaphone, Chronaphone, Vivaphone, Synchroscope, Cinetalk -- dozens and dozens of little companies started up to all try and push their way into the market. And they all tried the same thing and failed. They took a very orthodox approach. They tried to mate the phonograph with the silent movie to come up with something new, and it sort of worked, but not quite.

In 1925 a fellow named Lee DeForest had the bright idea to try and actually impress the audio on the same strip of media as the image. An optical soundtrack -- the very first one. He did quite well, except he was sort of a bumbling and lackluster entrepreneur, so he was unable to cut the key deals and get the kind of press that he needed in order to make a dent in the industry.

Eventually, of course, they figured out sound... -- but does anybody, by the way, know who actually solved the problem once and for all? Shout it out if you know...? It was AT&T -- the phone company. In 1925 they teamed up with Warner Brothers and decided to really lick the film sound problem, and in 1927 Al Jolson was singing away in the Jazz Singer, the very first talking film. There was a 50-year evolutionary period after that that got us through to THX. Right now where I think we are is at the beginning of the "talkies curve" with computers.

The computer that some friends and I built -- the NeXT machine -- is really like the Al Jolson of computing. And the question we should be asking is if Al Jolson is to NeXT, then THX and Lucasfilm quality Sound Design are to... what? Where are we going and what kinds of technologies do we need?

The more that people hammer away on rinky tink custom speech cards that all cost two or three thousand dollars, the more twisted and warped the approaches are going to get, and I think only by stepping back and tackling a hard problem are we going to be able to get the kind of integrated audio that we'll need to produce the most wonderful SIGGRAPH film, or the most wonderful user interface in the future.

So with that in mind, let me press a few buttons on the NeXT and just try and show you quickly the basic capabilities here. It's got high quality audio output, compact disc audio quality, and voice-quality input -- although you can also feed data directly into the DSP port and crunch on it. I think it provides some of the foundation tools that people will need if they want to invent say, the postscript for audio, or other such languages.

First of all, let's play two sounds real quickly. Here is some livestock I happen to have lying around; this is a cow. This is what a cow looks like. It's a frequency domain transform. Now I don't know of any computer yet that can tell the difference between a cow and a sheep, but if I bring up -- whoops, that's our cow again. Here's a goat. Goats are kind of like sheep. They bleat, in evenly-spaced pulses over time, and if I play the sound you'll hear that. This is the cow. (Moo!!) Whoa. And this is a goat. (Baaa!) Now it's real easy to tell at a glance which one is which. The cow has kind of a widespread spectrum with an amplitude and frequency curve that sweeps up; the goat does not. We can play the cow backwards. It still sounds like a cow -- kind of a distressed cow. On the other hand, as you might guess, a goat sounds the same forwards or backwards. It's a pretty symmetric sound in time. Whoops, that's a goat. Somewhere I have a human impersonating a goat. And this looks unfortunately different. There are still pulses, but they're not so well autocorrelated. It would be desirable to be able to come up with a language for representing a sound adequately so that you could tell the difference between sheeps and cows and goats and chickens, and people impersonating the same things, as well as different types of speakers. These sort of elements are not yet nuts and bolts that can be screwed into current user interfaces -- but they'll have to be.

NeXT, of course, has integrated audio into the mail system. I'll show that to you very quickly. You can send out a letter to someone. Let's send a note to Steve Jobs. I don't know if it will get there or not, but to send a little voice annotated message, we can record a little bit... "Hi, Steve, this is Mike. This is a test." And then play the sound back. ("Hi, Steve...") It's possible to go in and edit it. We see a little wiggly waveform and scroll around. Notice smooth scrolling everywhere. Select little bits of sound, play them out. I think there are lots of approaches that can be taken for this problem. Since we all are on such tight time budgets, let me come up to the microphone and sum up in about 30 seconds.

NeXT really is the first machine that you can purchase which has an "ear" (a microphone) and which has high quality sound output, and really the capability that one needs to approach the sound problem in a general way. I don't think it's been licked yet. We're just at the beginning of a very fun and very exciting curve, and I really believe that audio is going to impart more personality and spunk and feeling to user interfaces than we know what to do with right now. We need to think deep thoughts about more than just speech, and I hope that next year when SIGGRAPH rolls around there will be NeXT machines providing soundtracks for the movies -- not just squeaking Foleyed Luxos. Thanks very much.

Moderator
Chris Schmandt
MIT Media Lab

Thanks for bringing your toys, Mike. I wish we had more time to play with them. The final speaker is Barry Arons from Olivetti Research Center.

Barry Arons
Olivetti Research Center

Hi. I'm going to talk a little bit about a project that integrated voice and audio into the workstation, some hardware and software approaches of how we did that, and then some current research -- what we're doing now to improve upon that architecture.

The project I'm going to talk about is called the Conversational Desktop. It's built in a large extent upon the ideas in Phone Slave, that piece of tape that Chris showed earlier -- it was a conversational answering machine. However in the Conversational Desktop, there was a much deeper level of integration into a network of workstations.

One interesting thing that we did was to do audio-based direction sensing. The user would wear a headset mounted microphone

— ARONS - SLIDE 30 —
© Barry Arons

and behind him were two other microphones -- and just doing some simple level detection, we could determine if the user was talking in the direction of the workstation or talking to someone sitting in the office.

In and of itself it wasn't any great technology, but it was an exploration of unobtrusive techniques to turn the speech recognizer on and off, instead of having to have a manual switch, or to say "pay attention" and "stop listening" all the time.

I only mention it here to show that there is *lots* of potential uses for audio -- not just synthesis and recognition and recorded sounds -- in the user interface. People have to start thinking about ways of using audio.

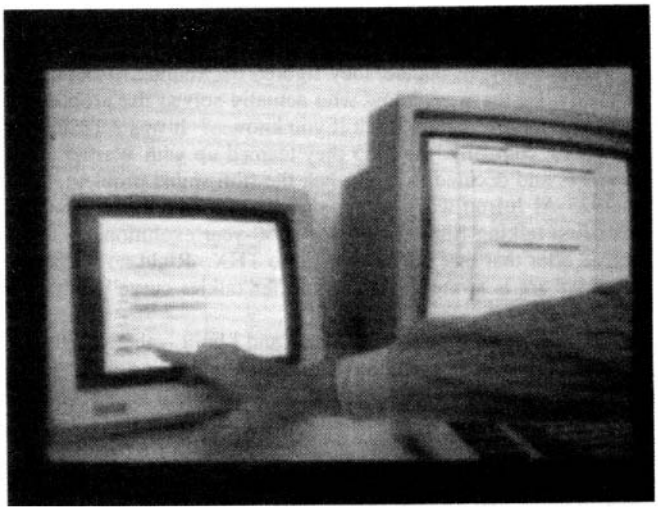
One last thing. If you'll notice in the background of that picture, there is about three dozen audio patch cords which we needed to set up the system and we'll get around to trying to eliminate those in a minute.

We used relatively simple audio and video teleconferencing. The audio just went over the telephone system. We had different kinds of reminders -- and voice mail. Since the machine had some idea of where you were, in your comings and goings, we could automatically change messages based upon if you're in your office, if you went out to lunch, and things like that -- automatically.

We tried to go *beyond* the desktop metaphor that's used in window systems, thinking about a *conversational metaphor*, where you really interact with the workstation by having a dialogue with it. We use dialogues for commands, for feedback to the user, and also as part of an error correction mechanism for the speech recognizer.

The real challenge was to seamlessly integrate multiple input and output media into the interface. On input we had voice, keyboard, touch sensitive screen, mouse, and touch tones from the telephone.

— ARONS - SLIDE 5 —
© Barry Arons



— ARONS - SLIDE 31 —
© Barry Arons

On output we had voice, text, graphics and video. And we had to come up with a software architecture to integrate voice into the interface.

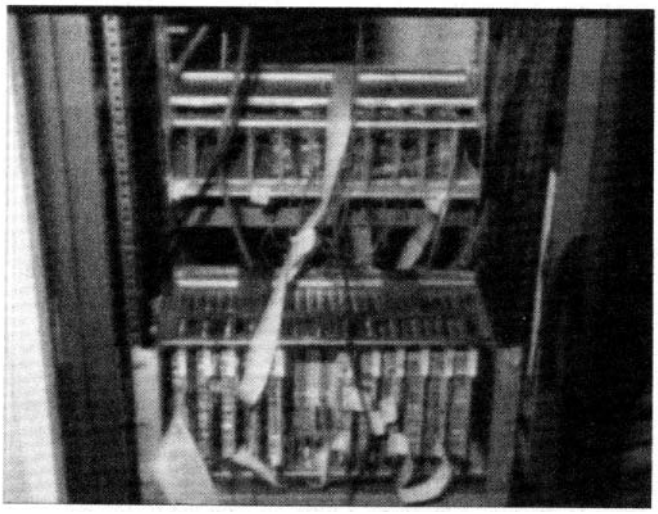
In a second I'm going to show a piece of video from the Conversational Desktop. This was our vision of what desktop audio -- a real desktop audio environment should be like -- and again, this is pretty old. This is from about five years ago. I'm sorry if you've seen it -- if not, the whole thing is in one of the SIGGRAPH video reviews.

— VIDEO TAPE BEING PLAYED —

Well, it's not quite like wearing a Walkman; maybe more like wearing this little toy we've got up here. You kind of get used to wearing it, but it gets in the way when you're trying to eat or something like that. (laughter)

Let me tell you a little bit about how we did the audio for the Conversational Desktop. We had a pretty simple server. It initially just did play and record, and interfaced to the

telephone. We later added speech synthesis and recognition. It was simple in that it was just a RS232 connection between a host computer and a dedicated PC, which we used as our audio peripheral.



— ARONS - SLIDE 28 —
© Barry Arons

The PC had a voice card on it, which did the play and record, and we just communicated over the serial link.

The advantage to this was that it allowed us to use *any* workstation as our host, and to have any kind of unusual hardware on the other end. This is actually the first sound system that we used. It was kind of a custom built thing, and you really don't want to plug that into every workstation that you want to use audio on.

The disadvantage of this set-up was that we only had a single hard-wired configuration. You couldn't re-route the audio easily. That's why we had to use the patch panel. Only a single application could use the audio hardware at a given time.

I'm a little hesitant to say this in front of a SIGGRAPH crowd, but what we really want is a server for audio that does what window systems have done for graphics. By that I mean a platform on which it's easy to build user interfaces to help bring interactive audio technology into the mainstream.

The graphics community has lots of experience with window servers, such as X or NeWS. What we really want to do is to take the techniques used in these servers, experience from things like the serial server that I mentioned, applications such as Phone Slave and Conversational Desktop, and really apply them into the audio domain.

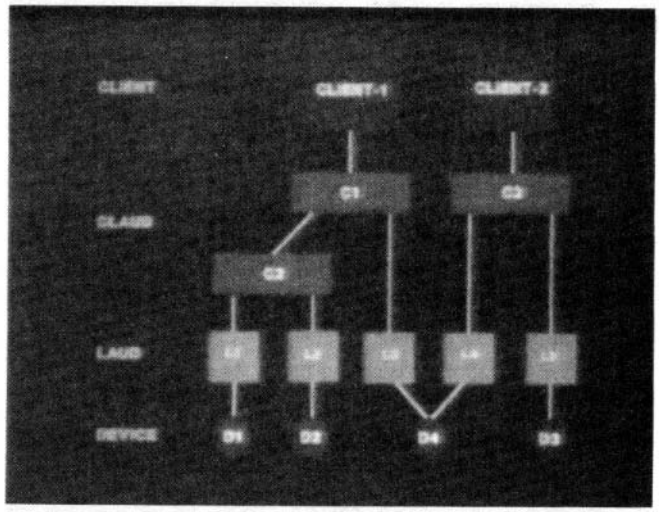
What we're doing at the Olivetti Research Center in Menlo Park is working on something called the VOX Audio Server. Like X or NeWS, you have a server that typically runs on your workstation, but it can run applications distributed throughout your network. It's multi-tasking in the same way that in a graphics system, such as X, you have a server in your local workstation, which draws onto the screen for multiple client applications. Here the audio server handles multiple audio requests from different applications.

We do all kinds of audio routing and mixing under completely under software control so that we can easily change our configuration. We might want to use recognition or synthesis, and change our configuration quickly, depending upon the application.

Like any good graphics system, we operate in a device independent manner so that we can support a wide range of devices without having to modify our application software.

Finally, we have a queueing mechanism that helps us reduce delays in real-time processing, something that's particularly a problem in UNIX systems. With the queueing mechanism we try to prefetch as much data as possible. If I know that I'm going to do a play and a record, I tell the server that that's going to happen sometime in the near future. The server prepares those as much as possible, possibly opening files, turning speech recognizers on and off, etc. Then we put the actual events in the queue to say play and then record. Then we can have transitions that happen as quickly possible. That either happens in the server -- if possible it happens in device drivers, and if possible, below that, it happens in the hardware -- if the hardware will handle it.

Here is a diagram of what VOX looks like.

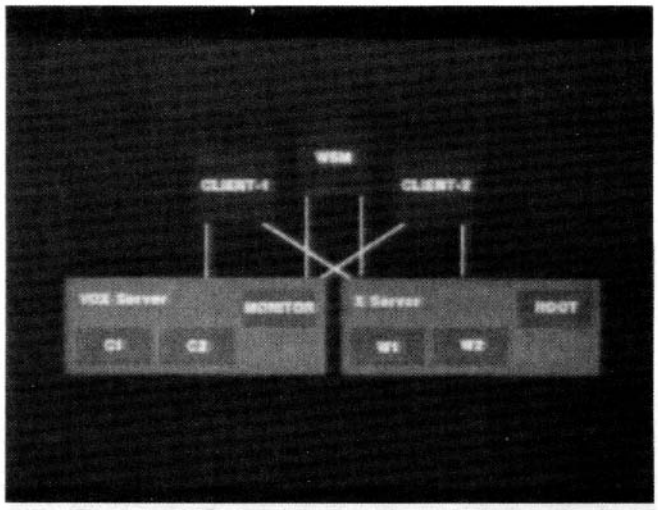


— ARONS - SLIDE 4 —
© Ing. C.Olivetti and C.,SpA.

At the lowest level are the actual physical devices. On top of the devices we have something that we call Logical AUdio Devices -- we call them LAUDS (pronounced *loud*) -- which are really the device independent abstractions on top of the hardware. Examples might be something that plays, or records, or synthesizes, etc.

These LAUDS have audio ports that we can interconnect under software control and we can combine the LAUDS together into useful audio circuits that we call Composite LAUDS or CLAUDS (pronounced *cloud*). These are created by the client applications and are controlled by them.

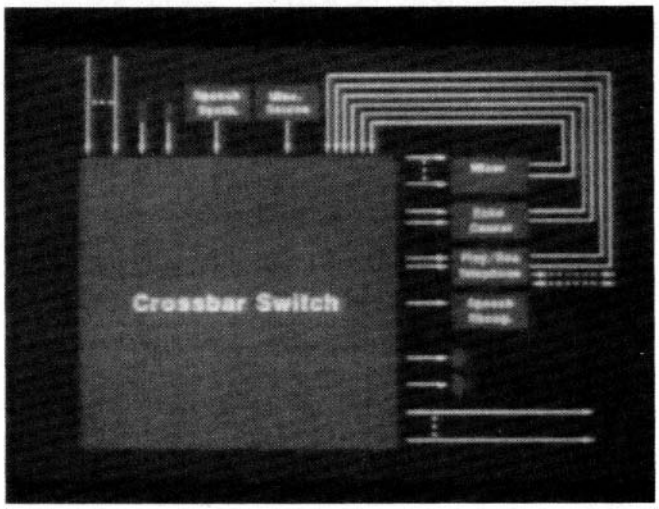
Here is a slightly higher level picture of how the server looks.



— ARONS - SLIDE 29 —

You can see how the audio server parallels the window server and it can work with it -- most applications are typically voice and graphics together. In the same way that a window system has a root window or a window manager that handles things like user preferences and input focus, we envision a similar entity for the audio server that oversees the resource sharing between clients -- sometimes these resources can be conflicting.

Here is what we envision our hardware environment to be -- this is what we're prototyping right now at Olivetti.



— ARONS - SLIDE 26 —

These are typically all analogue audio components around the outside -- synthesizers, recorders, playback, echo cancelers so we can do hands-free speaker phone applications, etc. In the center of all that is a big analogue crossbar switch, which really gives us the interconnectivity so that you can do all the interconnections under software control -- it eliminates *all* those patch cords.

So in winding up here, what are we going to do with this? Kind of everything that you've seen here on the panel today. It's really a toolkit that allows you to build applications. We see synergy between voice applications -- that just having a voice mail system on your machine or just an answering machine probably isn't worth the aggravation -- but it's really when you get applications that can share between them that you really see some benefit.

The current focus of our research is in using audio as a control mechanism, and for a communication channel, in a shared window system -- something that Keith Lantz mentioned yesterday in the multimedia panel. And we think that's a good step towards integrating voice and audio into the window system.

Finally, we're concentrating on VOX *functionality*. We're not worried too much about hardware -- we know we can put it on all digital hardware when it's available. What we're really trying to do is create tools for building desktop audio applications, which in turn will help create a market. Like Mike said -- we are really trying to encourage people to use and integrate voice and audio into the workstation -- just like text and graphics are today.

That's the end of my prepared presentation. What we're going to do now is breaking a little bit with the traditional panel format, is we're going to do some questions that we've prepared ahead of time, and then we're going to open it up to the floor as well.

ARONS: I'll first the first question address to Chris. Why hasn't voice been commercially successful in the workstation market?

SCHMANDT: My attitude towards this is that it hasn't been integrated with anything successfully. We've seen a number of voice workstations that have come and gone, but they're not the kind of things that I can edit files on, or compile on, or anything like that, and that's what I use my workstation for.

ARONS: The next one is directed towards Polle. What good is a multimedia document anyway?

ZELLWEGER: An obvious application of multimedia documents is in education. People can use two channels simultaneously -- the visual channel and the voice channel. For example, if an application can point in a document *and* simultaneously play an audio explanation, that's very good for explaining images and diagrams. You can also teach foreign languages, where it's really important to hear the audio. Or you can have articles about composers -- say, Beethoven -- that actually include snippets of their compositions. Another application is in more informal communication between people. Multimedia electronic mail would allow people to use voice for speed and expressiveness while retaining the advantages of asynchronous delivery and permanent storage.

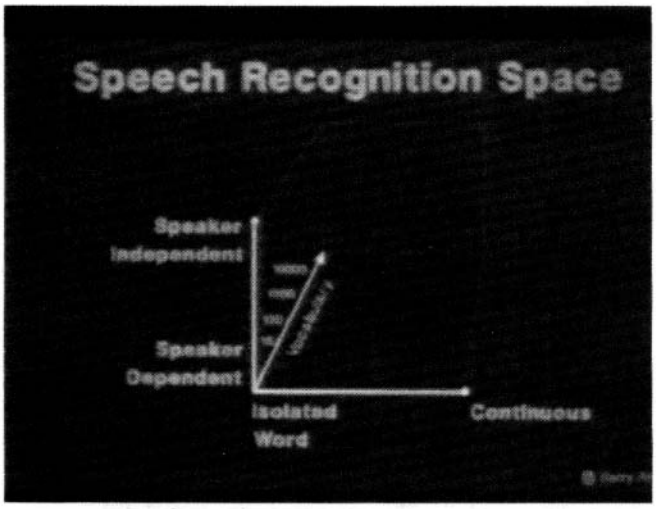
LUDWIG: I could add something to that too. I think that beyond some of the various isolated applications you mentioned there's this notion of "natural information" -- in nature there's redundancy across different kinds of channels, and there's also the notion that sometimes part of the information is just more easily conveyed in one format than another. Look at this conference, for example, there are slides, there are videos, there is audio... Think about having a conversation with someone. You use a blackboard, you'll draw pictures, you'll use your speech, all that sort of thing. Information just naturally takes different formats and often several at once.

HAWLEY: I have a quick comment to add. There is also always the question of what the appropriate media should be,

and sometimes the appropriate technology and appropriate media questions are bungled by people who want to make a splash. One example might have been -- well, I was the one who put Webster's Ninth Dictionary on the NeXT machine, and everyone asks "why doesn't it pronounce the words?" And although that seems on the surface like that would be a useful thing - - it probably will be in the fullness of time -- it's not appropriate for the current level of technology, you see, because there are only two ways to do it. You can have the computer leverage off of the pronunciation fields that are in the dictionary, in which case you get a DecTalk pronouncing Webster words, and that's not appropriate for a reasonably high quality dictionary. Or you could take the 60,000 words in the book, and have Edwin Newman be hired away from Simon & Schuster, read them all in -- it only takes about 40 hours. But at 4,000 bytes a word, let's see, 60,000 words is 240 megabytes. Now that's a lot of space. And although there's a company that did just that and pressed something very much like that onto a CD-ROM with a Voice of America narrator reading all the words, it's not clear right now that we can really tackle those problems -- at least, for consumers -- they are good research areas, but you need to be careful.

ARONS: Is speech recognition important to future window systems and what kind will be required?

Here is a diagram of the different kinds of speech recognition.



— ARONS - SLIDE 27 —
© Barry Arons

Speaker independent versus speaker dependent -- a question of whether you have to train the machine -- sit down and speak all the words to it. There is isolated word versus continuous. Continuous is what I'm doing now. I -- don't -- really -- like -- talking -- discrete. It gets to be a little bit of a problem if you really want to interact conversationally with your machine. And the third axis is really the size of the vocabulary. There's lots of different kinds of recognizers out there. Most of them tend along an axis -- if they're speaker independent and continuous speech, they have small vocabulary, etc., it's kind of hard. What you really want to be is way out in the far corner but there really aren't any good machines to do that yet.

SCHMANDT: That's sort of my attitude. I've just come from quite a period of time with trying to set up my voice navigation in X Windows so that four users could use it full time. And trying to set up a microphone system like this as opposed to one of these noise canceling head mikes -- it's not like wearing a Walkman. People won't use it. You end up with a lot of problems with acoustics of your room. Do people have the record player on? Do people have the stereo on? Does the user spit into the microphone when he talks? Does a user eat the mike? (gnaws on microphone -- laughter) Adjusting audio levels. I won't eat it again; I promise. That's what they told us at the beginning. They said please eat the mike. I didn't think they meant to take it literally.

Anyway, the long and the short of it is speech recognition is very difficult to use. So despite all of this hype about how the keyboard is going away and things like that, don't count on it.

ARONS: Just a couple quick slides which I could read, but they wouldn't do you much good. The "sun's rays meet" versus the "son's raise meat" -- clearly a problem for speech recognition. Here's another one -- "wreck a nice beach" versus "recognize speech". (laughter) Different kinds of problems. One's a question of synonyms and homonyms and the other's a question of finding the right word boundaries among other things.

HAWLEY: I have a quick comment. There are other aspects to speech recognition that aren't just mapping the sounds that come out of someone's mouth into text, which really haven't been tackled, and which might be appropriate sooner. I mean, general purpose speaker independent speech recognition is sort of a Holy Grail for computer science -- and when we see 40 MIPS and 40 megabytes as commonplace in workstations, I think we'll have the technology to crack it more or less. But until that time comes, what about some of the other problems, like should your workstation know whether or not it's you that's using it, or if you're a man or a woman. Well, if you were a man or a woman, and your machine could figure that out, maybe it could fix the pronouns in the documents for you; I don't know. But it seems clear to me that there are a variety of sound classification problems that have been unaddressed and which can afford interesting opportunities.

ARONS: This one's for Mike and Lester. What's the role of non-speech audio in user interfaces?

HAWLEY: I think the role for non-speech audio in interfaces is very much parallel to the role of non-speech audio in movies. It's a blank slate for us right now and kind of a golden opportunity. I was the first one at NeXT to make my text editor sound like a Three Stooges episode. "Pop" when window is closed and all that kind of thing. I think we'll see novelty sound effects for a while, but there are many, many opportunities to provide appropriate background effects for giving people much stronger indications of what on earth is going on in interfaces.

LUDWIG: I think sound effects are very important in lots of other ways besides movies, theater, and popular music. Some are using them to re-enforce some of events governed by window managers, like where you hit some window with your mouse cursor and the terminal "bings" at you in different ways reflecting different types of windows or application states. The idea here is to generalize the classic "beep" from your terminal when you or the machine do something.

There's also people using audio for more sophisticated things. One example from the scientific visualization area involves "clicking on" graphical surfaces with a mouse, or even dragging the mouse cursor along the graphical surface (as if you were "scratching" the graphical surface): in either case

you hear different kinds of sounds and attributes of the sounds tell you a attributes about the data set. People are investigating this sort of thing.

There's also other applications too where you could use environmental audio, for example in training an engine mechanic to listen for audio cues as running engines are adjusted and that sort of stuff. But for synthesized audio, a lot of times just having sort of backdrops of different kinds of sounds could be useful. Somewhere in this year's conference videos -- I forget which video -- someone showing something about some workstation or some data set, and just the mood that some of the background music created I think could be used as an audio symbol to denote the input focus in a multi-application display. Also, in Scott Fisher's stuff with the artificial realities, as you move from on artificial reality world to another, you could have different audio background themes cluing you in as to what world you currently were in. Such cues from backdrops can be very important I think.

ARONS: This is the last question -- if you want to get up -- we're going to take questions after this. What are the advantages and disadvantages of using analogue versus digital technologies?

SCHMANDT: Let me just say one word on that. Everything is digital down the road. Telephones are going to be digital. You think it's going to be a wonderful digital world. We have digital telephones at MIT and it ends up being a real pain in the something-or-other to try to get your computer to talk to this digital telephone -- even though it's all digital. There's protocol conversions, and you end up with a co-processor in your machine and suddenly to talk to your telephone line costs you an extra 1500 bucks.

ARONS: In terms of our set-up, we're really trying to set up a rapid prototyping environment so we can build applications like you saw here today really quickly and there just isn't the software running on a DSP that will let us simultaneously do recording and playback and synthesis and recognition all at the same time. So it's easier for us right now to use a bunch of analogue components and plug them together. It's really cheap and it's efficient.

The other problem with digital right now in terms of using external components is that there is lots of standards. There is all kinds of different encoding rates in bit rates for audio and different ways to compress it. So it's easier for us just to use analogue.

HAWLEY: I've always thought the question of analog versus digital is sort of a no-brainer -- the great property of all digital things is you can edit them, and once that door is open, you can do miraculous things with digital audio. But on the flip side there is a transition period that one has to get passed in order to ensure that the data is reliable. You see, the not so great property of digital things is that if a bit goes bad, the system tends to either work or not work. Namely, if a bit goes bad, it doesn't work. Analog media tend to degrade more gracefully. You accumulate noise, but at least you can hear some signal, whereas digital stuff is sometimes subject to drop-outs. So until the technology stabilizes, which also has something to do with standards and the amount of attention industry pays to it, I think digital is going to feel a little creaky to us, but it's clearly the right thing to do.

ARONS: We're going to open it up to questions -- go ahead in center.

Q. I have a really obvious question. Do we really want to be talking to our computers and do we really want to have this extra noise in our office space, because I get annoyed when I hear someone's Mac go "biqueeee" and all this irrelevant noise going on in my environment. I don't have the luxury of having my own office enclosed space isolating me and I don't necessarily want to wear a headset, so where do you see the happy medium there?

SCHMANDT: Good question, a very good question. The question is -- it has to do with appropriateness. If you're working in an environment in which you can hear other people, clearly you can hear them on the telephone. It's not an office if they're not on the telephone. I think hearing people having a conversation -- one side of a conversation on a telephone is a whole lot more distracting than hearing somebody occasionally speak the name of their window, which is what I'm seeing with some of my students right now. On the other hand, it's a real problem that voice does broadcast.

ARONS: In the center in the back.

Q. Hi. I'm interested in how you're solving problems with the general real-time operating system stuff, particularly relevant to I believe the NeXT and also Etherphone. Yesterday we talked in the multimedia session a lot about the problems of you've got no real-time operating system. How are you going to be scheduling between tasks, and not breaking up the signal that you're receiving from the user, or transmitting back. So I'm interested to hear how you're overcoming these problems, or whether you just ignore them and hope they go away.

HAWLEY: I think the operating systems question is really central to how adequately a computer can handle high bandwidth media. And current operating systems, as you point out, really cannot schedule audio efficiently enough in order to make the best use of it.

Now in the case of NeXT, we took a pretty straight-forward approach. We had to make the system be 4.3 BSD compatible, but we knew we were going to use MACH. MACH provides lightweight processing at the "thread" level, which is much more amenable to control of tasks like audio streaming through the machine. There are also some hardware innovations that facilitate this kind of thing, like the way the DMA works in the operating system -- it puts much less strain on the main CPU when some piece of sound goes squirting through.

I'm not trying to brush off the problem because it still exists and I guess in real practical terms the way the current NeXT machine scales up -- and remember the NeXT is only the next; it's not the last -- is that running off an optical disk one can generally play CD quality sound through the CPU and take down about 20% of the CPU. It's too difficult to record compact disk quality sound directly onto a magneto- optical disk because the writing bandwidth is not high enough on optical disk technology yet.

However, you can record stereo CD rate sound onto a magnetic disk -- but not much more than that. That's about as fast as the drives will go.

One last little seat of the pants number to keep in mind is that running at about 25 megahertz in an 68030, if you want to play telephone quality sound, which is commonly eight kilohertz and Mu-law-encoded -- that's 8,000 bytes per second with a special logarithmic type of encoding. The converters run at 44 kilohertz. And that means that you have to interpolate the sound -- which is an arithmetic-intensive computation -- in order to feed the DACs at the right rate. That's tricky. It can either be done through a DSP, which we have one of, or it can be done through the main CPU, which takes 20% or 40%. Either way, there is a lot of crunching that has to go on and it's very clear -- to me anyway -- that current

operating systems are sort of a shoehorn solution, but not really taking the bull by the horns.

ARONS: Over there; please state your name and affiliation so people know who you are.

Q. Nobody else had to. Bart Locanthi at Bell Labs. I have a flip side to the other question, which is are we sure we want our computers listening to us? I mean, when we think they're listening. Maybe they're listening when we're swearing at them or when we're --

HAWLEY: I'd kind of like to put an end to this line of questioning. Do we really want to read books off of computers? People ask me all the time. Why would I want to read Shakespeare off of a computer screen? And the point is that you guys are going to fix graphic display technology for us in the next 10 years and make it as compelling as paper. The same is true of color. Why would I want to look at anything on one of these obnoxious Mexican-color-TV displays that people are showing all over the place? People by and large do not make appropriate use of color technology. And I think audio -- both in and out -- is the same kind of thing. There's always the Big Brother potential looming in the background, but clearly there are very useful things one can do with sound. It's a virtually untapped resource and I think we have an obligation to figure out what the right things are to do with it before someone gums up the works by doing the wrong thing and selling too many computers.

SCHMANDT: In the back, please.

Q. Ken Pier, Xerox PARC. A question for Mike. What you demonstrated was a multimedia mail application and some isolated sort of manually done playback of pre-recorded voice or pre-recorded sounds. What kind of tools are available on the NeXT machine, or planned, to be able to build the kind of multimedia documents that we've been seeing?

HAWLEY: That's sort of a large question. Let me try and answer it real briefly. There is a fair amount of software for dealing with the digital signal processor at a nuts and bolts level. If you want to write auto correlators or do low level DSP stuff, you can do that. There is object-oriented stuff in a thing called the "sound kit", which controls management of audio in and out -- recording and playing of various formats. You can get down to samples and display them on the screen. That begins to tap into a software library called the Application Kit, which provides "view"-like objects and window objects to support display and editing of sound.

As far as integrating audio into documents is concerned, that's more of a can of worms that we've tried not to open up yet. A lot of people have done hyper-text very, very badly, and we've been quite conservative. You won't see "link" buttons in our documents yet. However, it is possible to nail in bits of sound and I think maybe Dick Phillips has done a little bit of that with his "living" SIGGRAPH proceedings demonstration. I haven't seen it yet, but -- high level integration we leave that to other people to figure out for the moment, because any other solution would be more of a liability in five years than a cure.

Q. My name is Mark Linnish, and someone talked a little bit about needing a language like Postscript for audio, and one of the things that really concerns me about this whole multimedia area is common file formats. How do you share those things and how do you share them over networks, and how can I receive a -- let's say a compound document with audio and some of these kind of things from my machine when you send it -- and we've got different machines and all that kind of stuff. So I was just going to ask the panel again about this idea of common formats or --

SCHMANDT: There's actually a lot of work being done on that. From my own personal point of view, the reason I've been using X windows is not because I love the X window system, but because of portability. As soon as I get a version of X running on the NeXT, then I can start playing with the NeXT.

In terms of interchange protocols -- you're talking standards, you're talking CCITT, you're talking about 10 years to get anything done. There is a lot actually happening in the X.400 series of protocols to sort of standardize some of those message handling, to at least allow us to have different body parts that have different media in it. It's still probably going to be a case that the speech format that I'm using on my machine may be different from the speech format that you're using on your machine, which may involve conversion and reconversion.

On the other hand, a lot of the speech coding came out of the need to get rid of a lot of bits and since my belief is that memory is cheap enough and disk space is plentiful enough that we may end up going with relatively unencoded speech just for the ease of moving it back and forth -- you know, 64 kilobit voice. You can edit it, you can do anything you want to it and it's relatively cheap to move around.

LUDWIG: I'd like to respond to that a little bit too, obviously because it's involved with communications. We're at a period now I think when there's a lot of change going on and usually when there's a period of change going on, there's some people who wish the change already happened and wonder why it hasn't happened yet and there's those who figure why even bother with the change. So all of us are sort of torn between what's comfortable and what's doable and what we know is feasible, and what we'd like to see and what we know we can do to get in between. And I think probably right now what's going on is there's some hesitation on the part of decision makers spending resources to develop these things in earnest, and the need to actually do that, the need to sit down with the problem and figure out what the appropriate machinery, what the appropriate technology is. It won't be until after some of that foundational work is done and you get the demonstrated possibility for market share that people will sit down and make the compromises that they need to get the different kinds of standards. Sure, there's the standards forms and so forth, but a lot of times they -- with all due respect -- work in isolation from the technology and the best technical solution.

So I think until there gets to be some unification as to what we want to do with this stuff, what's the right kind of thing to do and what sort of the common denominator is across a lot of the applications, the so-called primitives -- maybe that's what the Postscript thing is. It's going to be a bit of a problem to expect the standards to come prematurely, and I think we're seeing that in all kinds of things -- not only digital audio or the audio functionality, but also in the HDTV business that's going on in this country, if you're familiar with that. Everybody's trying to figure out whether it should be like a TV set or more like a workstation. Same kind of phenomena.

HAWLEY: Just to sum up with a real quick analogy, it strikes me that your question is a great one, but it's asked really early. It's a little bit like asking "What about Postscript?" before they made a laser printer. Right now there's only one computer that does general audio in and out, and it's the NeXT. Maybe there will be more; I hope there would be lots more to help create that industry.

LINNISH: If the standards were there though, I could envision buying one of these things for fairly cheap and I can't do that right now.
HAWLEY: Well, I think it's the cart before the horse though. Standards come after you push enough devices and technologies around.
LUDWIG: We also know about standards that are made prematurely. I won't mention any of them.
SCHMANDT: Over here on this side.
Q. Cliff Bashears, Columbia University. My concern is on voice as an input device. I haven't heard any mention of using the pitch to control say a valuator, and in general a more rich taxonomy of devices as you find in the graphics literature. Have you people thought about how to categorize different parts of the voice and simulate other devices that are commonly used, and do you think that's a hokey idea or do you think you should incorporate it in VOX? I just want to hear your thoughts about it.
SCHMANDT: I've done a lot of work of trying to understand human intonation, just of speech, not of non-speech sounds. It's very exciting, it's very important. It ends up being extremely difficult to do.
HAWLEY: Go talk to Bill Buxton about that one. He knows lots of people who've already controlled sliders by singing various pitches into a microphone. And in my own lab when I do music research I often sing notes into a pitch tracker, to provide pitch input, or to push other things around. It's moderate hokum, I would say. But there might be something there, particularly for handicapped users.
ARONS: At H.P. Labs we started doing a little bit of work to try to use pitch to help speech recognition. If you can tell by the pitch if something is a question or a statement, you can do better in the recognition phase.
HAWLEY: But I could put on my Media Lab hat there for a second too, which is to say one of the things that people do when they listen is detect anxiety or pleasure and other features like that in the voice of the person they're talking to, and it might be useful at some point in time if the computer can tell when you're screaming at it -- notice whether or not you're distressed and maybe offer some assistance. (laughter) So in the very long term, there are a number of let's say more emotional features that could be measured, and which would be useful.
ARONS: Again, over here on the right; I'm afraid we're running out of time -- so this is going to be the last question.
Q. I'm Leo Hourvitz from NeXT also, and I want to grouse about telephony and see if we've got any way out of this. I sat about 10 feet away from the geek when he and Barry did the Phone Slave, and I thought this was really great. Gee, I want one of these bad. Now we can see machines coming out that like have the ability to throw that audio around. I mean, eight kilobytes per second isn't that bad on these generation of workstations. But the phone interface has gone backwards. I mean, it used to be you knew what a phone was. You know, Bell came, they wired it in, it was analogue. You could get by with two wires if you had to. But now, those things -- I haven't had a desk with a normal phone in like four years, right. It's all been some kind of PBX which has changed about every eight months. And the interface to every PBX is different. I don't have an analogue phone line any more. How can I possibly connect my computer to it? Okay, panel, get us out of this.
SCHMANDT: That's what ISDN is supposed to be all about.
HOURVITZ: Right. But is ISDN going to make it to my desk or is it going to stop at the PBX which would still be one of 10 different brands.
LUDWIG: It's going to your desk.
SCHMANDT: It will go to your desk, but not to your home.
HOURVITZ: Gee, great.
LUDWIG: I don't know if I would agree with that one, but I guess I'm...
HAWLEY: I don't use phones. I use E-Mail now.

Moderator
Barry Arons
Olivetti Research Center

We're running out of time. There's another session which is coming in here very shortly, so we're not going to have time to answer any personal questions here -- if people would go to the breakout room next door...

We'd like to thank the Panels Committee and SIGGRAPH for allowing us to do this. I'd like to thank the rest of the panel, particularly the co-chairman Chris -- and we'd like to thank everybody for showing up. We hope that it sparked your interest in audio, and perhaps this session will help in integrating audio into the workstation. Maybe a few years down the road we'll all see you at the ACM SIGAUDIO conference or something like that.