1992

# Voice technologies in advanced computer systems

Jodie M. Cone
*University of Northern Iowa*

# Voice technologies in advanced computer systems


## Abstract

Recently there has been a surge of interest using voice technologies in advanced workstations. The motivation for voice is its role as the primary channel of human-to-human communication, which ties in with current research in which computers are used to facilitate group problem solving, in enhanced user interfaces, and office computing. Taken broadly, the use of speech as a command and data channel may require digital recording and playback techniques, speech recognition, text-to-speech synthesis, and telephone interface equipment. The big payoff will be to build systems, using these technologies, to allow computers to become a part of the infrastructure of daily human communication (Schmandt and Arons, 1985).

# Voice Technologies in Advanced Computer Systems

A Graduate Paper

Submitted to the

Department of Curriculum and Instruction

In Partial Fulfillment

of the Requirements for the Degree

Master of Arts

UNIVERSITY OF NORTHERN IOWA

by

Jodie M. Cone

August 1, 1992

This Research Paper by: Jodie M. Cone

Entitled: Voice Technologies in Advanced Computer Systems

has been approved as meeting the research paper requirement for the Degree of Master of Arts

Sharon E. Smaldino

July 20, 1992
Date Approved

Director of Research Paper

Robert R. Hardman

July 17, 1992
Date Approved

Graduate Faculty Adviser

Robert R. Hardman

July 17, 1992
Date Approved

Graduate Faculty Reader

Peggy Ishler

July 22, 1992
Date Approved

Head, Department of Curriculum and Instruction

Table of Contents

# Chapter 1

# Introduction

## Background

Recently there has been a surge of interest using voice technologies in advanced workstations. The motivation for voice is its role as the primary channel of human-to-human communication, which ties in with current research in which computers are used to facilitate group problem solving, in enhanced user interfaces, and office computing. Taken broadly, the use of speech as a command and data channel may require digital recording and playback techniques, speech recognition, text-to-speech synthesis, and telephone interface equipment. The big payoff will be to build systems, using these technologies, to allow computers to become a part of the infrastructure of daily human communication (Schmandt and Arons, 1985).

Computer-based workstations are already much better suited than telephones for improved user interfaces. They have superior displays, typically CRT screens, in part because of power requirements and cost, telephone displays are small and difficult to read. Workstations have better input devices, with full keyboards and a mouse. More importantly, a variety of powerful window

systems provide rich and flexible graphical interfaces (Schmandt, Arons and Simmons, 1985).

## Purpose

A large number of potential audio applications such as conversational answering machines, unified electronic mail systems, and interactive audio training systems, have been under development. It is a strong belief of Schmandt and Arons (1985) that "although no single voice utility may be overwhelming in isolation, a synergistic collection of multi-media applications making appropriate use of voice will provide a very powerful communications environment" (p.22). This paper will focus on these voice applications and if they will have a place in everyday life at work and home.

## Definitions

Algorithm: A defined process or set or rules that leads and assures development of a desired output from a given input (Sippl and Sippl, 1980, p. 13).

Audio Bridging: The means by which speech is recognized by the computation device.

Commands: An electronic pulse, signal, or set of signals to start, stop or continue some operation (Sippl and Sippl, 1980, p. 90).

Data Channel: The bi-directional data path between the input/output devices and the main memory in a digital computer that permits one or more input/output operations to happen concurrently with computation (Sippl and Sippl, 1980, p. 128).

Digital Signal: Commands in the form of digital signals which are sent to the processor to be exercised out.

Duplex Echo Canceler: The simultaneous verification of information sent between two computers.

Interface: A common boundary between automatic data processing systems or parts of a single system (Sippl and Sippl, 1980, p. 255).

Kbits: One thousand binary digits (Sippl and Sippl, 1980, p. 273).

MACH: Operating system environment use for Macintoshes.

Multi-way Audio Conferencing: Computer conferencing with the use audio between more than two persons.

Parser: A routine that controls decoding of an external program statement by establishing its syntactic tree, according to the specified syntax of the programming language (Sippl and Sippl, 1980, p. 379).

Processor: A device capable of receiving data, manipulating it, supplying results usually of an internally stored program(Sippl and Sippl, 1980, p. 405).

Pulse Code: A code which sets of pulses have been assigned particular meanings (Sippl and Sippl, 1980, p 429).

Syntax: The rules governing sentence structure in a language, or statement structure in a language such as that of a compiler program (Sippl and Sippl, 1980, p. 544).

Texas Instrument Speech Card: Adapter card that converts digital information to speech.

Transfer Rates: The speed at which data may be read from or written to the device, from the lowest to the highest speed and density available (Sippl and Sippl, 1980, p. 593).

UNIX: Portable operating system environment that is more widely used in scientific environments.

User Interface: The bridge between the user and computation device.

VAX: Large mainframe operating system environment in which terminals are connected.

Voice Mail: Way of storing speech.

Workstation: Self-contained computation area.

X Window Systems: A window environment that runs under UNIX.

# CHAPTER 2

## Literature Review

### Uses of Voice Technologies

There are a range of applications in which one could incorporate voice. For example, voice may be used to annotate text, as editorial comments on a manuscript or as part of an on-line tutorial. Voice may be incorporated in a more general multi-media document, such as a repair manual or video-based educational system. Voice, either synthesized from text or pre-recorded, may allow remote telephone access to databases, such as electronic mail, flight departures, or up-to-the-minute stock quotations. As far as currently used successful applications, voice mail takes the lead (Schmandt and Arons, 1985; Slowiaczek and Pisoni, 1982).

Listening typewriters, voice annotation of text, interactive audio training systems, voice mail systems, and computer conferencing all have potential use for voice applications. Other potential uses include using the telephone to access data, speech substitutes for the mouse and keyboard, auditory icons, speed dialing tools, and answering machines. (Schmandt and Arons, 1985).

Voice mail systems are computers that intercept calls and digitize speech, though they don't yet provide a means to access the stored voice from the rest of the office computer environment. A greater variety of services, conference, forward, transfer, pickup, etc., are becoming more widely available. The addition of control interfaces opens the possibility for workstations to perform intelligent call setup and routing using these services (Schmandt, Arons and Simmons, 1985; Schmandt and McKenna, 1988).

Computer-supported teleconferencing is gaining popularity. It has the potential to take two forms. One would simply consist of sharing monitors and setting up a separate telephone call, the other would be to add a computer-mediated voice link. The data and voice links can be initiated using a single conference management application. The voice can be used to initiate changes in who has the "floor" and the teleconference, including the voice, can be logged. Multi-way audio conferencing can be realized with the telephone network and conventional audio bridging equipment, or by computer networks that transmit the voice digitally (Schmandt and McKenna, 1988).

Speech recognition is a more difficult problem. Despite significant media hype, the general purpose large-vocabulary "listening typewriter" is far from being available; keyboards will be with us for the foreseeable future (Schmandt and Arons, 1985). Recognition would be very useful over the telephone, but reduced audio bandwidth and noise problems make this one of the more difficult feats in the field. Specific workstation applications might be amenable to voice input, especially if the mouse is being used for other functions, such as drawing lines in a paint program. Speech can be used simultaneously with mouse input for control or to drive menus. At the Massachusetts's Institute of Technology (MIT) Media Lab an alternative approach is exploring voice to move between windows; speech is used as the channel for the "meta-dialog" of communication with the window manager, rather than just a keyboard substitute (Schmandt and Arons, 1985).

Audio also tends to be a difficult medium, and it is suggested that it be used as a secondary, or simultaneous interaction. For example, since voice is more useful when one's hands are busy, it would be beneficial to use while running a CAD system or driving a car (Hayes and Reddy, 1983). The use of voice

computer applications is more tolerable when no other access mechanism is possible, such as for someone who is physically impaired.

## Motivation

A system called the "Conversational Desktop" is located in the Media Lab at MIT. This system is designed to use voice to interact with telephone, scheduling, airline reservations, and audio memoranda functions (Schmandt and Arons, 1985).

Since speech is a natural way of communicating, it is the basis for voice interaction with the office workstation. An office has major telecommunication needs which are in a large part voice oriented, and voice may be used in dictating or conversing with co-workers face to face (Schmandt, Arons and Simmons, 1985).

Speech was also employed because of its ease of use, it was desirable to use connected speech recognition because of its tolerance for more natural spoken input. However, connected speech recognition is more difficult than isolated word recognition, due in large part to difficulties finding word boundaries and variation in the pronunciation of words in an acoustical context (Oshika, 1975).

To use connected recognition effectively, an application must take the output of the speech recognizer, consider it as a very noisy signal due to errors, and filter out the errors from the words which were correctly recognized. At that point, a variety of techniques may be used to try to insert the missing words or interact with the user to complete the transaction (Levinson, 1978).

The approach MIT took to solve this problem was to use a parser. A parser controls the decoding of an external program according to the program language. This particular parser was designed specifically for speech and the errors produced by a connected speech recognizer. The parser communicates with a dialog generator capable of phrasing a series of questions in order to gather enough correct input to perform the user's request. For recognition MIT employed an NEC DP-200 connected recognizer with a 150 word vocabulary (Schmandt and Arons, 1985).

The parser is unique in that it not only analyzes speech input based on a formal description of the syntax of the set of input utterances, but also is designed to detect input errors and parse the remaining correct

sentence fragments (Schmandt and Arons, 1985; Schmandt, Arons and Simmons, 1985).

Two major software components make up the system at MIT (Schmandt and Arons, 1985; Schmandt, Arons and Simmons, 1985). These consist of a higher level that extracts subsets of the tokens and calls the lower level repeatedly to test them against the grammar. A scoring metric is employed to select that subset which is most likely to correspond to what was spoken.

The lower level component consists of a set of grammar rules. This level simply applies a set of rules to analyze its input. While output is being analyzed from the recognizer, a frame-like representation of the input is built up, which includes an indication of which slots in the frame are missing to complete the command. The lower level parser returns information that is used by the scoring metric in the higher level (Schmandt and Arons, 1985; Schmandt, Arons and Simmons, 1985).

When an insertion error occurs, spurious tokens appear in the output from the speech recognizer (Levinson and Rosenberg, 1978). Substitution errors replace correct tokens with incorrect ones. In both cases, detection of incorrect tokens is preferred so as to continue processing only the tokens believed to be

correct. This can be accomplished by considering all substrings of the recognizer output, and selecting the best by a scoring algorithm.

An example given by Schmandt and Arons (1985) if the string ABCD were returned from the recognizer, the following substrings would be considered: ABCD, ABC-, AB-D, A-CD, -BCD, and so on. Each substring is analyzed according to the grammar to determine whether it is syntactically correct. For each syntactically correct substring, a score is computed to determine the most likely match between the input and what the user intended.

Any substring, according to Schmandt and Arons (1985), whether a sentence or sentence fragment is a possible candidate. A candidate is chosen by applying a scoring metric based on the following:

1. Completion: a complete sentence is preferred to a fragment, as one is more likely to speak a complete command to the machine.
2. Number: of two possible substrings, the one with the larger number of tokens will be selected.
3. Adjacency: additional weight is given to adjacent tokens. For example, if the original input was ABCD,

the substring ABC- has a higher adjacency score than AB-D.

Since a significant portion of the problem for connected recognition is segmentation, adjacency is a powerful measure for connected speech. If it is determined that the second token in a sentence is correct, it is more likely that the first and third tokens will also be correct because at least one of each of their boundaries must have been determined correctly (Rabiner and Levinson, 1981; Zue, 1985). This implies that if ABC is a complete sentence, then all substrings including A-C are considered sentences, and scored using the same metric. The reason being that the correct acceptance of AB if the recognizer returns an incorrect sentence ABX. Even though AB is incomplete, it is an accurate indication of a portion of the speaker's intent and should guide further dialog (Schmandt and Arons, 1985).

## Computer Knowledge

A phrase in the vocabulary is defined as a "particular instance of a small number of syntactic categories"(Rosenberg and Schmidt, 1979 p. 1804). For example, "Jodie" is an instance of the category NAME, and "place a call" is an instance of a category

CMD_NAME; a command which requires a person's name for completeness. In general, this grammar was structured such that the category to which a command belongs indicates the number and types of the arguments to the command. For example, "place a call" would be recognized as a single utterance, the same as "Jodie." Basically, the number and types of arguments for an instance depends on the category to which a command belongs.

Commands are categorized by the types and numbers of their arguments which allows the parser to incorporate semantic knowledge as well as syntax, and also conveniently reflects the level of description used for the recognizer (Aho and Ullman, 1977). For example, if a call came in and someone said, "Schedule a meeting with Jodie," the recognizer should match against two templates, "schedule a meeting" and "Jodie." So the 'CMD_NT'(a command from the computer language developed for the conversational desktop) rule applied to "Schedule a meeting with Jodie" sets the 'command' field to MEET and the 'name' field to JODIE.

In addition to rules for complete sentences, the grammar also contains rules for incomplete sentences. The speech parser will try to recognize fragments such

as "Jodie tonight" otherwise it will make a rejection error, and return such a fragment to the processor, even if the user spoke a complete sentence (Aho and Ullman, 1977). It is then up to the substring generator and scoring metric to accept the best choice as no further processing of input will be done.

It is possible, however, to fill in the slots of an incomplete command through pre-existing knowledge in the system. This is done by applying simple rules which extract information from the context of the dialog and current state of the system. Use of this information can reduce requirements on the speech recognition hardware and tends to make the system more conversational (Schmandt and Arons, 1985).

An interesting example given by Schmandt and Arons (1985) is illustrated by the "When is my flight?" command. If a destination city is not explicitly stated, the next airline reservation that occurs in the calendar is reported. However, if the user is on the phone with a person who lives in another city, the schedule is first scanned for flights to that city (unless there is a flight departing imminently, in which case it is reported).

## System Requirements

When voice systems were first experimented with, much time was devoted to data compression schemes, with a goal of cutting transfer rates while minimizing the impact on intelligibility. As memory sizes, computer power, and network speeds have increased, the need for many of these techniques is decreasing. Evolving applications tend to use 64 Kbits/second log PCM (pulse code modulation) or 32 or 16 Kbits/second encoding (such as ADPCM, Adaptive Delta PCM) to obtain telephone quality speech (Martin, 1989). This results in continuous data transfer rates within the ability of most workstations and computer systems.

A digital signal processor (DSP) is likely to be required for both recognition and synthesis. This approach may be more cost effective in the future allowing rapid change between synthesis and playback by simply downloading a new algorithm. This same DSP may be also used for other communication tasks including modem or facsimile emulation. For example the TI speech card supports recognition, recording, and text-to-speech synthesis, and the Natural Microsystems Watson card supports modem signaling in addition to recording (Martin, 1989).

An audio environment must allow multiple processes to access audio hardware and files; these processes themselves may be distributed (Aho and Ullman, 1977). Mechanisms must be provided to detect conflicting requests for scarce resources and to arbitrate between them. Graphical interfaces must be closely coupled with sound interfaces, capable of maintaining synchronization during record and playback activities (Hayes and Reddy, 1983). A means of representing multi-media objects in the context of user selections must be provided; this will be most useful if it allows maximum "interoperability" with current text-only operations.

A server can be used to provide synchronization. There may be a lapse though between the time when a request is submitted until it can be processed because it has to wait on the que (Martin, 1989).

The MIT audio server has been most noted for its emphasis on the requirements of an audio user interface and the projects it was associated with; the Phone Slave and Conversational Desktop. Although it did not include any routing primitives, it did implement event ques and synchronization. MS-DOS provides a convenient real-time environment, and speech boards for the PC bus are plentiful and inexpensive. This voice server was

designed for use by a single computer (Schmandt and Arons, 1985).

## Limitations

Several limitations in utilizing audio in computer systems can be blamed on the medium, while some on current technology (Levinson and Rosenberg, 1978; Rosenberg and Schmidt, 1979). All of which must be taken into consideration when designing applications and user interfaces. Speech is slow; at 150 words per minute it is much slower than our ability to read from screens or paper. It is also "bulky" to store at thousands of bytes per second, and it cannot be scanned like text. Listening, especially to synthetic speech, puts a cognitive load on the user that can interfere with other types of mental activity.

As for currently available devices, there are limitations on the intelligibility of synthetic speech and the error rates of recognizers. For synthesis, text is broken down by the application of several layers of linguistic rules into sound units called phonemes, which are then realized as an acoustic waveform. Text-to-phoneme rules break down for many proper nouns, and phoneme realization problems make some sounds easily confused (Oshika, 1975). According to Schmandt and

Arons (1985), "Speech recognition is still in its infancy, and even small-vocabulary speaker dependent devices have difficulty in acoustically imperfect environments without noise-canceling microphones worn on the head. Recognition is generally easier on isolated words than on continuous speech, and when used in a speaker-dependent rather than a speaker-independent manner" (p.21).

It has been suggested that limitations may be reduced by translating the temporal nature of speech into a spatial dimension, possibly providing cues like time (tick marks) and speech/silence intervals (color, size), in addition to a cursor that moves as the sound is played (Schmandt and McKenna, 1988). This means give speech some sense of place inside a voice document or voice file being edited. Because it is so slow, audio playback needs to be interruptible, and mechanisms should be provided to re-play, speed up, or skip ahead. Transitions between modes must be done with minimal overhead, such as when switching between play and record modes in a conversational system or stopping the playback of prompts as immediate feedback when a caller pushes a button on a telephone keypad. Speech recognition is especially difficult in the user

interface, requiring careful choice of vocabulary, application, and crafting of functionality; speech recognition as a direct keyboard replacement almost never works (Martin, 1989).

The parser used at MIT was designed to cope with only one speech recognizer; the NEC DP-200 and as such is constrained by many limitations (Schmandt and Arons, 1985). For example, the lack of any second guess information for each word or sentence, and the lack of any measure of the recognizer's confidence in its selection for each word are noted. Also, inadequate subsetting; allowing recognition to be limited to a particular set of words at any moment to improve recognition is a problem. Another limitation tends to be no indication of the relative difficulty of discriminating between various words in the vocabulary.

# Chapter 3

## Conclusion

A large number of potential audio applications such as conversational answering machines, unified electronic mail systems, and interactive audio training systems have been under development. Is there a place for voice technologies to be used in our everyday environment?

With the use of these new innovations, the world is becoming a global village making it faster and easier to obtain information that may have taken weeks or months in the past. As the world becomes a more enhanced network, the opportunity to become more efficient and effective becomes more important.

By incorporating voice technologies into domestic life, quality and efficiency can be improved upon in workstations in the office as well as the home. The conversational answering machine is a good example of improved quality. By allowing the calling party to interact with the answering machine a detailed message is given.

Although it may be a while before some of these technologies reach their potential, there are a few innovations that are being used quite extensively today.

For example, voice mail is being used by major corporations all over the world. This makes it possible for people to receive messages from any location in reach of a phone. This is not only more effective, but efficiency is enhanced by reducing the amount of messages that would have to be taken at the corporate office, and by reducing the probability of lost messages.

Other technologies that are still in the developmental stages, such as the unified electronic mail system, have made great advances. Electronic mail messages are being integrated with voice messages and may be viewed on the screen or heard over the phone with a text to speech synthesizer (Schmandt and Arons, 1985). The fact that several limitations exist does not discourage researchers from overcoming the barriers to not only improve the quality, but also to make the technologies more accessible.

Schmandt and Arons (1985) feel that "voice is a demanding medium, but clearly one that is ripe for integration into our everyday computing environment. At present there is a high demand for interactive voice applications; unfortunately the medium is difficult to work with and the currently available technology has

many weaknesses. An integrated environment using voice (both as data and control) over a number of applications suggests a server-based approach. Although current technology can support such a server, its architecture is still a research topic"(p.24).

References

Aho, A.V. & Ullman, J.D. (1977). Principles of Compiler Design. Addison-Wesley.

Hayes, P. & Reddy, R. (1983). Steps Toward Graceful Interaction in Spoken and Written Man-Machine Communications. International Journal of Man/Machine Systems, 19, 231-284.

Levinson, S.E. (1978). The Effects of Syntax Analysis on Word Recognition Accuracy. Bell System Technical Journal, 57(5), 1627-1644.

Levinson, S.E., & Rosenberg, J.L. (1978). Evaluation of a Word Recognition System Using Syntax Analysis. The Bell System Technical Journal, 57, 1619-1626.

Martin, G.L. (1989). The Utility of Speech Input in User-Computer Interfaces. International Journal of Man/Machine Systems, 30, 355-375.

Oshika, B.T. (1975). The Role of Phonological Rules in Speech Understanding Research. IEEE Transactions on Acoustics, Speech, and Signal Processing ASSP, 23(1), 104-112.

Rabiner,L., & Levinson, E. (1981). Isolated and Connected Word Recognition--Theory and Selected Applications. IEEE Transactions on Communications, 25(5), 621-659.

Rosenberg, A.E. & Schmidt, C.E. (1979). Automatic Recognition of Spoken Spelled Names for Obtaining Directory Listings. The Bell system Technical Journal, 58(8), 1797-1823.

Schmandt, C., & Arons, B. (1985). A Conservational Telephone Messaging System. IEEE Transactions on Consumer Electronics, 30(3), 21-24.

Schmandt, C., Arons, B., & Simmons, C. (1985). Voice Interaction in an Integrated Office and Telecommunications Environment. Paper presented at the meeting of the American Voice Input/Output Society.

Schmandt, C., & McKenna M.A. (1988). An Audio and Telephone Server for Multi-Media Workstations. Proceedings of the 2nd IEEE Conference on Computer Workstations (pp. 150-159).

Sippl, C.J., & Sippl R.J. (1980). Computer Dictionary (3rd ed.). Indianapolis: Howard W. Sams and Co. Inc.