HUMAN-COMPUTER INTERACTION, 1989, Volume 4, pp. 67-94

The SonicFinder: An Interface That Uses Auditory Icons

William W. Gaver
University of California, San Diego, and
Apple Computer Inc.

ABSTRACT

The appropriate use of nonspeech sounds has the potential to add a great deal to the functionality of computer interfaces. Sound is a largely unexploited medium of output, even though it plays an integral role in our everyday encounters with the world, a role that is complementary to vision. Sound should be used in computers as it is in the world, where it conveys information about the nature of sound-producing events. Such a strategy leads to auditory icons, which are everyday sounds meant to convey information about computer events by analogy with everyday events. Auditory icons are an intuitively accessible way to use sound to provide multidimensional, organized information to users.

These ideas are instantiated in the SonicFinder,[1] which is an auditory interface I developed at Apple Computer, Inc. In this interface, information is conveyed using auditory icons as well as standard graphical feedback. I discuss how events are mapped to auditory icons in the SonicFinder, and illustrate how sound is used by describing a typical interaction with this interface.

Two major gains are associated with using sound in this interface: an increase in direct engagement with the model world of the computer and an

---

Author's present address: William W. Gaver, Rank Xerox Cambridge EuroPARC, Ravenscroft House, 61 Regent Street, Cambridge CB2 1AB, England.

---

[1] The SonicFinder, Finder Sounds, Finder, and SoundManager are trademarks of Apple Computer, Inc. As to SonicFinder, copyright 1987, 1988, 1989 by Apple Computer, Inc.

CONTENTS

added flexibility for users in getting information about that world. These advantages seem to be due to the iconic nature of the mappings used between sound and the information it is to convey. I discuss sound effects and source metaphors as methods of extending auditory icons beyond the limitations implied by literal mappings, and I speculate on future directions for such interfaces.

1. INTRODUCTION

Imagine what life would be like without sound. Many events would go unnoticed—some pleasurable, like birds chirping in nearby trees; some useful, like hearing a person entering the room behind you; and some crucial, like hearing a car approach as you cross a street. If given the choice, most of us would prefer to hear the world around us. Why should it be any different for the world of the computer?

The appropriate use of nonspeech sounds has the potential to add a great deal to the functionality of computer interfaces. Just as there are occasions in which pictorial displays (i.e., icons, windows, etc.) can convey information more concisely and intuitively to users than writing, so may there be situations in which sounds can provide information better than speech or visual displays. In Gaver (1986), I outlined what seemed to be a promising approach to using sound in the interface. In particular, I advanced a strategy for developing *auditory icons,* which are everyday sounds meant to convey information about events in the computer by analogy with everyday events. In this article I elaborate and update that argument and show how it may be applied by describing the SonicFinder, an auditory interface I developed at Apple Computer, Inc. This interface demonstrates a number of the characteristics of auditory icons that make them an appealing addition to the interface designer's repertoire, and raises a number of issues about their use in the future.

Before describing the SonicFinder, I present three arguments for why sound should play an integral role in computer interfaces. Then I briefly discuss the idea of auditory icons and the theoretical perspective upon which this strategy is based. After this foundation is laid, I describe the SonicFinder in some detail and illustrate the ways sounds are used in this interface by examining a typical interaction with it. Finally, I discuss why the mappings used in this interface seem intuitively obvious and speculate about the future of auditory interfaces.

2. WHY USE SOUND?

Until recently, the use of sound in the interface has been largely confined to providing auditory alerts to users. Many people find these bleeps and buzzes distracting and irritating. If this is the case, why should sound play a larger role in the interface?

2.1. Because It's There . . .

Buxton (1986) speculated that if anthropologists of the distant future were to find a computer system of the 1980s, they would conclude (among other things) that humans must have had a "low-fi" ear. Current uses of sound are extremely limited in the functions sound perform and in the manipulations of sound that are employed. Clearly, a good first reason to use sound is simply because it is there. Audition represents another sensory modality for people, one that has not been exploited in today's visually oriented interfaces.

Not only is hearing "there" in most people, but sound is increasingly "there" in computers as well. The greater processing speeds and memory capabilities of today's machines enable the sorts of sound production and manipulation that used to require specialized hardware and software found only in major research laboratories. The new generations of personal computers make fairly sophisticated uses of sound practical: Indeed, the auditory interface described in this article can be run on any of the Macintosh family. Of at least equal importance for the future of sophisticated auditory interfaces is the advent of the Musical Instrument Digital Interface (MIDI), the standardized protocol for communication between computers and more specialized digital music devices (Loy, 1985). Using MIDI to control external signal-processing equipment allows designers to overcome the signal-processing limitations of current personal computers.

In summary, sound should be used in the interface because hearing is a largely untapped modality for people and because sound production is a seldom-exploited resource of computers. However, sound is more than just an available resource. Sound plays an integral role in our everyday encounters with the world, one that is complementary with vision. It is in understanding this role that the most compelling reasons for using sound become clear.

2.2. We Use Sound in Everyday Life

The promise of nonspeech audio as a display medium stems partly from the fact that most of us rely on sound for information in our everyday lives. We listen to the thunk of a car door to find out if it has closed properly, to the gurgle of pouring liquid to know if a container is almost full, and to traffic noises to assess the danger of crossing a street. Mechanics listen to automobile engines, and doctors to heartbeats, both with the aim of getting information about mechanisms that are not visually accessible.

Many people also listen to their computers to get information about mechanisms that cannot be seen. They listen to their disk drives to find out whether data is being accessed properly, to their printers so they know whether a misfeed must be corrected, and to their modems to find out whether a proper connection has been achieved. It is a paradox of the computer age that although sound has been designed out of most systems, people rely a great deal on those sounds that remain.

In proposing sound as a display medium, the notion is that we can start putting useful sounds back into computers. But instead of being accidental results of hardware engineering, such sounds can be explicitly designed to be useful. Understanding the relations between sound and vision can help explain why this is a worthwhile endeavor, as well as the situations in which audio will be a particularly valuable addition to graphics.

2.3. Listening Complements Looking

Most people both listen to and look at the world. The reason for this is not only that using both senses increases the bandwidth of available information, but because sound and vision are complementary modes of information. A simple way to contrast listening and looking is to say that although sound

***Figure 1.* Complementary modes of sound and vision.**

	TIME	SPACE
SOUND	**Sound exists in time.** • Good for display of changing events. • Available for a limited time.	**Sound exists over space.** • Need not face source. • A limited number of messages can be displayed at once.
VISION	**Visual objects exist over time.** • Good for display of static objects. • Can be sampled over time.	**Visual objects exist in space.** • Must face source. • Messages can be spatially distributed.

exists in time and over space, vision exists in space and over time (see Figure 1).

Sound exists in time: It is an inherently transient phenomenon. Sounds have a beginning and an end, and most sounds are brief enough that both are experienced. Most visual objects, on the other hand, tend to persist: Their creation and destruction are only occasionally witnessed. This means that sounds are well suited for conveying information about changing events (e.g., closing doors, pouring liquids, and approaching cars), and vision for information about relatively stable objects. Conversely, although auditory information is usually available for a limited amount of time, visual information is usually available for repeated sampling (I might not hear a sound again, but I can look at an object more than once). Interactions with visual objects are more flexible over time, but they also tend to produce more clutter than auditory events.

Vision exists in space: In order to take advantage of visual information, one must look in the appropriate direction. Sounds may be heard from all around: One does not have to face a source of sound to listen to it. This implies that sound can convey information to users despite their orientation, whereas visual information depends on users' directed attention. On the other hand, many visual items can be displayed simultaneously in different locations. There is probably a much lower limit to the number of auditory messages that can be presented at once.

Sound and vision are not only complementary informational media, they also convey different sorts of information. Vision largely depends on the reflection of light from surfaces, whereas sound is caused by vibrating materials. So sound can provide information that vision cannot, for instance about occluded events or the internal mechanisms of complex objects. We can hear events in the next room, or listen to automobile engines, even if we cannot see them. In the interface, it may often be more appropriate to use sound in providing information about background processing or events in occluded windows than to invent visual metaphors.

In general, auditory displays have the potential to convey information that is difficult or awkward to display graphically. Sound can provide information about events that may not be visually attended, and about events that are obscured or difficult to visualize. Auditory information can be redundant with visual information, so that the strengths of each mode can be exploited. Using sound can help reduce the visual clutter of current graphic interfaces by providing an alternative means for information presentation. Finally, auditory interfaces may help make the increasingly spatial interfaces of current systems accessible to visually impaired users (see Edwards, this issue). The complementary nature of sound and vision is the last—and I believe most powerful—reason for creating auditory interfaces.

3. EVERYDAY LISTENING AND AUDITORY ICONS

If sounds are to be used in the interface, they should be used much as they are in our everyday lives. Other methods for creating auditory displays that have been suggested (Blattner et al., this issue; Bly, 1982; Edwards, this issue; Mansur, Blattner, & Joy, 1985; Mezrich, Frysinger, & Slivjanovski, 1984) are based on using variations of sounds (e.g., in pitch, loudness, or timbre) to differentiate messages. These systems are often quite promising, but mapping information about computer events to musical variations is usually somewhat arbitrary. In my view, it is better when possible to map the attributes of computer events to those of everyday sound-producing events. This is the philosophy behind auditory icons, which convey multidimensional information about computer events by analogy with everyday events. To understand why I think this strategy is likely to be a useful one, consider how people listen to the world in their everyday lives.

3.1. Everyday Listening

Traditional accounts of the psychology of hearing approach sound in one of three ways: in terms of sound's effects on the auditory system, with the aim of explaining music, or in order to measure the effects of environmental noise. Usually sounds are analyzed with respect to the physical attributes of the sound wave, which correspond fairly well to the fundamental attributes of music. Thus, the literature is full of accounts of pitch and loudness perception, the perceptual dimensions of timbre, and the relative effects of octave equivalence and pitch contour on the recognition of tunes. As psychologists, we tend to talk about sound as if it were always either music or noise.

But such an understanding does not seem satisfactory in explaining how we use sounds in our everyday lives. We do not hear the pitch of closing doors; instead we are more likely to hear their size, the materials from which they are made, and the force used to shut them. We do not hear the fluctuations of loudness in the sounds made by pouring liquids, but instead whether the liquid is viscous or fluid, how fast it is pouring, and how full the receptacle is. In general, we do not seem to hear sounds, but instead sources of sound.

The experience of hearing sounds per se is one of musical listening, whereas that of hearing attributes of sound-producing events is one of everyday listening. Although much more is known about musical listening than everyday listening, several studies have explored the experience of everyday listening. For instance, Vanderveer (1979) played recorded tokens of 30 common sounds such as clapping and tearing paper to subjects in a free identification task. She found that subjects tended to identify the sounds in terms of the objects and events that caused them, describing their sensory qualities only when they could not identify the source events. Ballas and his colleagues (Ballas, Dick, & Groshek, 1987; Ballas & Howard, 1987; Ballas & Sliwinski, 1986) studied ambiguous environmental sounds: They found that the time it takes to identify a sound depends on the number of possible sources it might have. Finally, in a protocol study I ran (Gaver, 1988), subjects were asked to describe tokens of 17 everyday sounds; like Vanderveer (1979) I found that these descriptions were of sources and that misidentifications were based on similarities among source events. In general, these studies all support the reality of everyday listening as an experience of sound sources, quite distinct from the experience of sounds per se.

Note that this perspective does not imply a distinction between everyday and musical sounds, but between the experiences of everyday and musical listening. It is possible to listen to most sounds as one does to musical notes—to attend to their pitch, loudness, and timbre—and to listen to the ambient auditory environment as one does to works of music, attending to rhythms, pitch contours, and the like. Conversely, often one listens to musical pieces with the aim of discovering the instruments involved, or how many performers there are. Strictly speaking, only the distinction between the two kinds of experience is valid.

Nonetheless, I refer to everyday and musical sounds in the discussions that follow. Although we can usually experience a given sound in terms of its source or in terms of the properties of the sound itself, many sounds encourage one form of listening more than the other. We seem more likely to notice pitch when listening to a continuous, unchanging, synthesized tone, for instance, than when hearing the dynamically changing sound produced by shattering glass. Noticing the source of the shattering sound, on the other hand, seems

***Figure 2*. Attributes of everyday listening.**

Perceptual Attributes	*Effects on Soundwave*
Material	
Restoring Force (e.g., hardness, tension)	Overall Frequency
Density	Frequency
Damping	Frequency dependent amplitude functions of partials; frequency
Homogeneity	Complex effects on spectrum and amplitude function
Configuration	
Shape	Spectral pattern, frequency
Size	Bandwidth, frequency
Resonating Cavities	Changes amplitudes of partials; may be primary determinant of sound quality
Interaction	
Type (e.g., hitting, scraping)	Overall amplitude function, spectrum
Force	Bandwidth, amplitude

much more compelling than that of the synthesized sound. In distinguishing everyday and musical sounds, the emphasis should be on the kind of experience a given sound affords, whether it is one of the sound itself or of its source. In terms of the interface, the distinction is one of the dimensions of sound that are used to represent data.

Sound is produced by the interaction of materials at a location in an environment. If the dimensions of musical listening correspond to fundamental physical attributes of the sound, the attributes of everyday listening correspond to the attributes of the source. People may be expected to listen for auditory information regarding the type, force, and duration of interactions; the type, size, and configuration of materials; and the size and kind of environment in which sound-producing events take place. Figure 2 contains a list of the kinds of attributes I hypothesize are important for everyday listening, and suggests the attributes of the soundwave that might be responsible for their perception. This figure is meant as an alternative to the traditional lists of sound attributes such as pitch and loudness, and may be useful in suggesting the kinds of attributes that can be manipulated to create auditory icons.

Several studies have explored how sound conveys information about the physical characteristics of sound sources. Warren and Verbrugge (1984) showed that the perceptual distinction between breaking and bouncing bottles depends on temporal patterning of spectrally identical sounds. Freed and Martins (1986) studied the perception of mallet hardness from the sounds of struck objects. They identified a number of acoustic correlates to mallet hardness that seem to determine a listener's perception of this event. Finally, I studied (Gaver, 1988) how people hear the material and length of struck wood and metal bars. This work suggests the attributes and acoustic correlates of sonic events shown in Figure 2.

An important property of everyday sounds is that the information they convey is multidimensional and organized. A single sound can potentially provide information about many different attributes of its source. For example, the sound of a slamming door might provide information about the size and material of the door, the force with which it was shut, and perhaps the size of the room in which the door is closed. These attributes are organized simply because sound-producing events have a structure. Attributes like size, weight, and material apply to the door, whereas others, like force, type, and duration, apply to the kind of interaction. Different sounds might be produced by different interactions involving the same door, or by different doors undergoing the same kind of interaction. Sonic events can thus be distinguished along some dimensions while remaining the same along others. This fact makes everyday sounds particularly suitable for mapping to events in the computer.

3.2. Auditory Icons

Everyday listening, then, is the auditory perception of the attributes of everyday events. Auditory icons exploit people's tendencies to listen to sources by mapping attributes of everyday sound-producing events to attributes of the model world of the computer. In the SonicFinder, for instance, selecting a file makes the sound of an object being tapped. The type of file is indicated by the material of the object, and the size of the file by the size of the struck object. Auditory icons of this sort are similar to visual icons in that both rely on an analogy between the everyday world and the model world of the computer. The selection sound, for instance, is based on an analogy between the event in the computer — selecting a file — and an everyday event — tapping an object. The mappings between other attributes of the computer and everyday events are clear once this basic analogy is understood. Because they exploit the power of such organizing metaphors, auditory icons may be expected to be as easily learned by users as visual icons are. In addition, because listening and looking provide complementary kinds of information, auditory icons can be created that will both complement and supplement visual icons.

In particular, it should be possible to create auditory icons that represent the objects and actions of the computer world in an intuitive way, simply by mapping them to the objects and interactions of everyday sound-producing events. The appropriate mappings should be obvious: Objects in the computer world should be represented by the objects involved in sound-producing events; actions by the interactions that cause sound; and attributes of the system environment (e.g., processing load or available memory) by attributes of the sonic environment (e.g., reverberation time). When auditory icons are based on such analogies, the relations between them and the information they are to convey should be obvious.

Multidimensional information can be conveyed by auditory icons if the dimensions to be displayed are mapped to dimensions of everyday events, such as the force of an interaction or the size of an object. In this way, any one sound can convey a great deal of information. Moreover, "families" of auditory icons can be created by exploiting the organization inherent in everyday events. For instance, if the material of a sound-producing event is used to represent the type of object, all auditory icons concerning that type of object would use sounds made by that kind of material. So text files might always sound wooden, whether they are selected, moved, copied, or deleted. In this way, a rich system of auditory icons may be created that relies on relatively few underlying metaphors.

Further benefits may be realized when the same analogy underlies both auditory and visual icons. The increased redundancy of the interface should help with learning and remembering the system. In addition, making the model world of the computer consistent in its visual and auditory aspects should increase users' feelings of *direct engagement* (Hutchins, Hollan, & Norman, 1986) or *mimesis* (Laurel, 1986) with that world. The concepts of direct engagement and mimesis refer to the feeling of working in the world of the task, not the computer, and are closely related to the notion of a transparent interface. By making the model world of the computer more real, one makes the existence of an interface to that world less noticeable. Providing auditory information that is consistent with visual feedback is one way of making the model world more vivid. In addition, using auditory icons may allow more consistent model worlds to be developed, because some computer events may map more readily to sound-producing events than to visual ones.

The idea of using auditory icons in the interface is an appealing one largely because it is based on the way people listen to the everyday world. If a model world can be used to represent events in the computer (as it is when visual icons are employed), then good auditory icons can be created simply by using the sounds that would be produced by that model world. Such auditory icons should represent multidimensional, organized information in an intuitive way. They can provide information that visual displays do not, and thus extend the consistency of the model world. Finally, using auditory and visual icons together should help create a more encompassing world for the user.

4. THE SONICFINDER

These ideas are instantiated in the SonicFinder, an interface in which information is conveyed using auditory icons as well as standard graphical feedback. This interface is meant to address several questions about auditory icons:

- Can everyday sounds be found that map naturally and meaningfully to events in the interface?
- Will auditory icons be useful and acceptable to users?
- When is sound a particularly appropriate display medium? That is, in what situations can sound convey information that graphics cannot, or in a form that is more appropriate than graphics?

Finally, developing and using a working auditory interface was expected to raise—and has raised—new issues that are not immediately apparent without such an example.

The SonicFinder is implemented in the form of functions called from within the original Finder[2] (at the time of this writing, version 6.0), which is the top-level interface to the Apple Macintosh. The Finder is the application that is automatically run when the Macintosh is booted: Organized by analogy with a desktop, it provides a visual representation of the items of interest in the interface (e.g., files, folders, disks, etc.) and allows users to manipulate them (e.g., to move, copy, or delete files).

The SonicFinder uses the information that is available in the existing interface to trigger and control the playback of sounds sampled from recordings of everyday sound-producing events. Thus, the SonicFinder extends the visual desktop metaphor into the auditory dimension, and does little else that changes or adds to the essential events and objects of that model world. This interface was felt to be an appropriate domain for implementing auditory icons because it is a well-defined direct manipulation interface, because it has comparatively general functionality, and because it is used relatively often by users of the Macintosh.

4.1. Sound-Producing Events in the SonicFinder

In the SonicFinder, events in the interface (e.g., selecting a file) are mapped to sound-producing events (e.g., tapping an object). Figure 3 summarizes the events in the SonicFinder that are mapped to sounds.

Computer events are defined as actions upon an item. Items are divided into *objects,* which include files, applications, folders, disks, and the trashcan, and *windows.* The actions that can be performed upon these items, such as selection, dragging, opening, or scrolling, are listed below each class of item.

---

[2] The SonicFinder exists in two forms: as a special version of the Finder in which the auditory portion of the interface is inherent to the Finder, and as an "init" file in which the SonicFinder code resides in a separate file called *Finder Sounds.* In the latter form, the auditory portion of the interface is only called if the init file is in the System Folder when the system is booted.

Figure 3. **Mapping events to sound in the SonicFinder.**

Finder Events	*Auditory Icons*
Objects	
Selection	Hitting sound
Type (file, application, folder, disk, trash)	Sound source (wood, metal, etc.)
Size	Frequency
Opening	Whooshing sound
Size of opened object	Frequency
Dragging	Scraping sound
Size	Frequency
Where (windows or desk)	Sound type (bandwidth)
Possible drop-in?	Selection sound of disk, folder, or trashcan
Drop-In	Noise of object landing
Amount in destination	Frequency
Copying	Pouring sound
Amount completed	Frequency
Windows	
Selection	Clink
Dragging	Scraping
Growing	Clink
Window size	Frequency
Scrolling	Tick sound
Underlying surface size	Frequency
Trashcan	
Drop-in	Crash
Empty	Crunch

Each item–action combination forms an event that is represented by a basic sound-producing event. Thus, object selection, the first event in Figure 3, maps to tapping an object; dragging an object is represented by scraping a surface; opening a window makes a whooshing sound; and scrolling makes a clicking sound.

Secondary attributes of these sound-producing events are used to convey further information about the computer events they represent. For instance, when an object is selected, its type (i.e., whether it is a file, folder, etc.) is represented by the material of the sound-producing object, so that files make wooden sounds, applications sound like metal, and other object types make other kinds of sounds. The size of the object is also conveyed by the sound, so that large objects make lower pitched sounds than small objects (as they do in the everyday world). These secondary attributes are shown in italics below the major mappings in Figure 3. In general, any one sound in this interface may convey several kinds of information, so that the sound made when selecting a file not only confirms the basic event, but also provides information about the type of selected object and its size.

Giving the feel of an auditory interface in a written article is obviously a difficult task. Listing all the auditory icons of the SonicFinder is only of limited use, because the way these sounds work with each other and with visual feedback is hard to imagine. Therefore, I instead describe a typical interaction with this interface in some detail.

4.2. A Typical Interaction: Dragging a File to the Trashcan

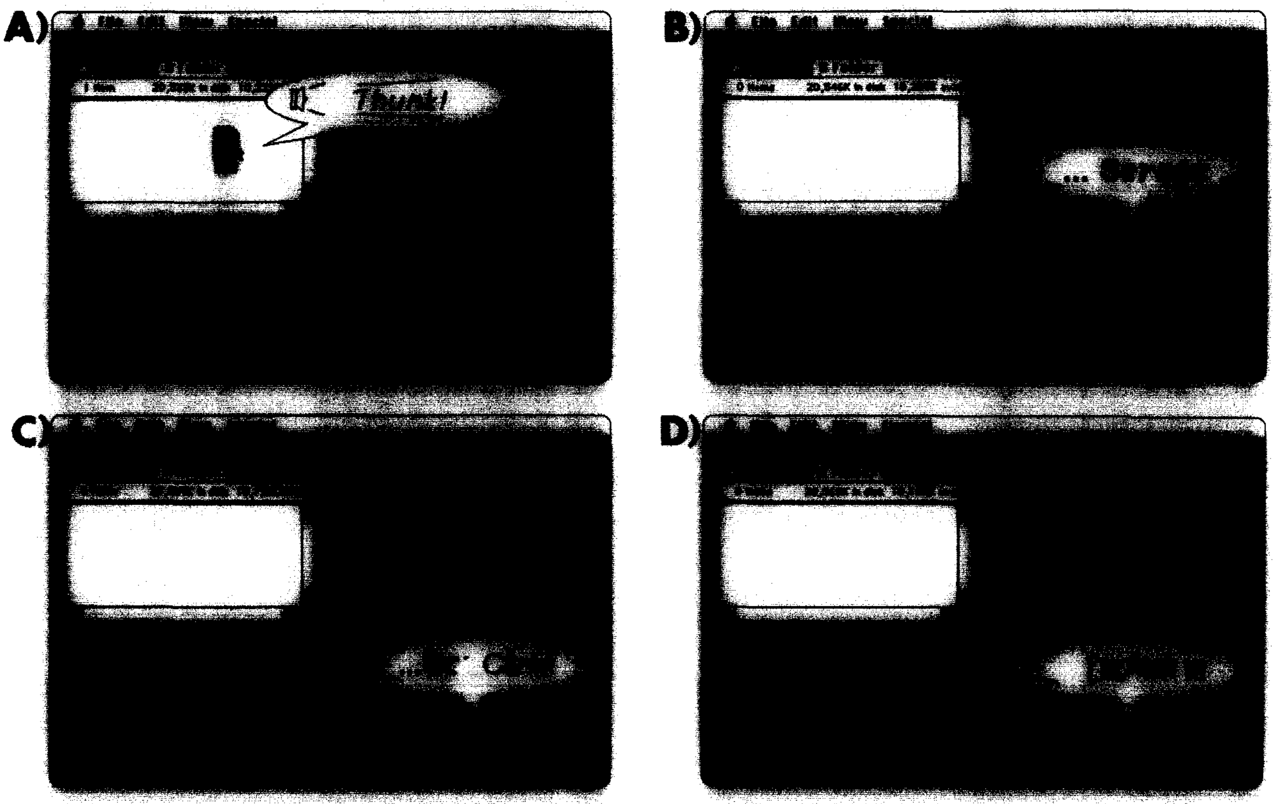
Figure 4 shows the series of events involved in dragging a file to the trashcan to delete it. First, the user selects the file (Figure 4A). This is indicated both visually, by the file becoming highlighted, and aurally by the sound of an object being tapped. The type of object is conveyed by the material being tapped. In this example, the object is a file, so it makes a wooden "thunk." If it had been an application, it would have made a metal sound; a folder would have made a sharper paper-like sound; disks a hollow metal sound (like a large metal container being tapped); and the trashcan a different hollow metal sound. In the Finder, there are standard icons for folders, disks, and the trashcan, but applications and files are not distinguished by icon type. These are easily differentiated in the SonicFinder by the use of different sounding materials for their selection sounds.

The size of a selected object is also indicated by the selection sound, so that large files or folders with many items in them make lower sounds than small files or nearly empty folders. This mapping of frequency reflects the physics of sound-producing events, in which large objects typically make lower sounds than small ones. Disks make sounds proportional to the available space in them, so that hard disks typically make lower sounds than floppies, and disks that are empty make lower sounds than disks that are full. Note that visual information about the size of objects is not typically available unless requested by the user. In this example, such information is only available from the auditory icon.

Once the user has selected the file, he or she can drag it toward the trashcan (Figure 4B). An auditory icon is also associated with this event—in this case, a simple scraping sound is played continuously while the object is being dragged. As with selection, the frequency of the scraping sound depends on the size of the object being dragged.

The scraping sound also conveys information about where the object is dragged. If the object is over a window it makes a sound with fewer high-frequency components than if it is over the desktop. If it is moved from the window (or desktop) where the dragging operation was initiated, then another bandwidth change is approximated by changing the sound's fre-

Figure 4. A typical interaction with the SonicFinder: dragging a file to the trashcan. (A) The user selects the object and hears the impact. (B) The user drags the object and hears it scrape along the desktop. (C) The user drags the object over the trashcan; the scraping sound stops and the trashcan makes its noise. (D) The user releases the object; a satisfying crash provides confirmation of the deletion. Typical interactions with the SonicFinder produce many sounds which are varied according to parameters of the objects involved. These sounds not only provide information about the events, but tend to make the world of the computer more "real" to users.



quency by an octave. This produces the impression of a similarly pitched sound with a different high-frequency makeup and is used to conserve the memory necessary to store new sounds. These bandwidth changes are meant to convey the sense of different surfaces over which the object is dragged and are a start toward conveying the sort of information necessary to discriminate whether dropping the object would result in a move or a copy. Ideally, of course, this information would be directly mapped to the sounds so that dragging sounds would change depending on whether a move or copy was implied, but for reasons of practicality such a mapping was not possible in this implementation.

When the object is dragged over the trashcan (Figure 4C), the scraping sound stops and the trashcan makes its selection sound. More generally, whenever an object is dragged over a container into which it can be dropped (e.g., a folder, disk, or the trashcan), the scraping sound stops and the container's selection sound is played. Auditory confirmation that a target has been hit turns out to be one of the most obviously useful features of the SonicFinder, especially in finding small folder icons that may be partially obscured by overlapping windows. A common problem in hitting such targets comes when the object, but not the cursor, is positioned over the target. In this situation, dropping the object does not place it inside the target, but instead positions it so it obscures the target further. The auditory icon indicates a true hit, and so reduces the amount of time spent playing "chase the trashcan."

Finally, when the object is dropped (Figure 4D), the sound of shattering dishes is played. This sound provides satisfying feedback that the object has been successfully marked for deletion. When the deletion actually occurs, a crunching sound is played to indicate the destruction of the object.

This sample interaction illustrates a number of aspects of the SonicFinder. First, many sounds accompany the use of this interface, so that even such a simple sequence as trashing a file involves at least four sounds. This means that a fairly constant ambient auditory environment is created when using the SonicFinder, so no single sound is particularly incongruous or distracting. Second, the sounds seem to fit well with the events they represent. The analogies between computer and everyday events are obvious, and the sounds used seem intuitive and natural (I explain why this is so in Section 6). This extends to the mapping between secondary qualities of the events, so that size and location are easy to encode and interpret. Finally, even in this example auditory icons convey information that the graphic portion of the interface either does not display (e.g., about file size), displays less effectively (e.g., about dragging over a container), or in a less satisfying way (e.g., about trashing). In the following sections, I discuss the lessons to be learned from the SonicFinder at some length.

5. ASSESSING THE SonicFinder

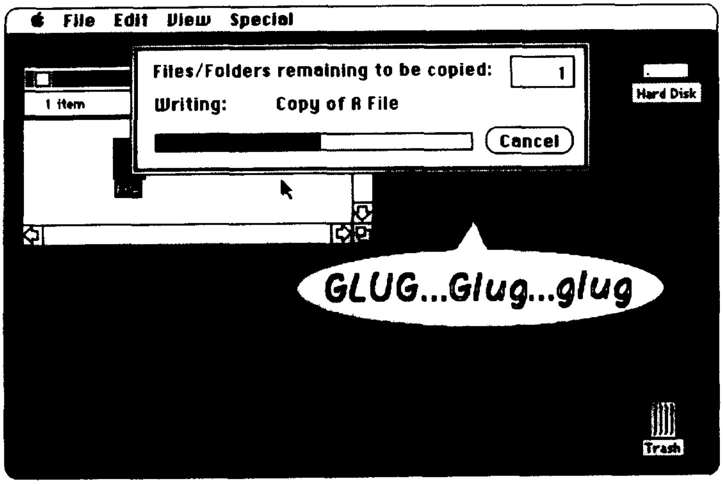
The SonicFinder is a working system and has been distributed informally within Apple Computer, Inc. For technical reasons, current versions do not work well on all machines. Nonetheless, there are a number of people, including myself, who use it as their standard interface as of this writing, more than a year after it was developed. One of the most telling pieces of evidence in favor of the addition of sound in this interface is that users complain of missing it when they use a quiet Finder. For these people, at least, the addition of sounds is valuable, and for this reason alone, the SonicFinder must be counted as a success.

Similarly, when the SonicFinder is demonstrated, the audience reaction is typically quite favorable. It seems that little justification is needed for the addition of sounds. The ways sound works within the interface seem natural and obvious to those who encounter it.

This is so despite the fact that the SonicFinder is a somewhat limited example of what auditory interfaces might be like. The auditory icons used in the interface seldom present valuable information that is not already effectively conveyed by the graphic portion of the interface. When auditory icons do convey new information (e.g., about the size of files) it is not clear that this information is necessary. So, with exceptions discussed here, the major function of sound in this interface is to provide redundant information to users. Provision of redundant information is likely to be an important role for sound, both in allowing more flexible and less attention-demanding interactions with the interface and in increasing feelings of direct engagement with the computer. But equally important is the largely unexplored potential for sound to convey relevant information that is not effectively conveyed by visual means.

The use of sound in the SonicFinder is limited partially because of constraints on the manipulations of sound that could be made. The auditory portion of the interface was implemented using the SoundManager, a recent addition to the Macintosh Toolbox (Apple Computer, Inc., 1988). The SoundManager made the implementation of the SonicFinder a relatively simple endeavor, but also constrained the sounds that could be used. In particular, sophisticated manipulations of sound, such as those involving filtering or reverberation, cannot be made by the SoundManager. In addition, technical problems less related to manipulations of sound were also encountered in implementing the SonicFinder. The most serious of these relate to the amount of memory required to store and play the sounds. Sampled sounds tend to be quite large (16 sampled sounds are used in the current version; their average size is about 14 Kbytes). Reducing the number and sizes of the sounds used was an important constraint on the design of the SonicFinder.

Figure 5. **Copying is indicated by the sound of pouring water; the frequency of the sound is continuously increased to indicate its process by analogy with the sound a container makes as it is being filled. In current versions of the Finder, the progress of the copy operation is also visually displayed using a "dial bar," shown in the upper center of the figure.**



Files/Folders remaining to be copied: 1

Writing: Copy of A File

Despite its present limitations, the SonicFinder is successful in demonstrating some of the potential of auditory icons. Two major advantages of this interface are apparent after using it. First, the addition of auditory icons increases feelings of direct engagement with the model world of the computer. Hearing as well as seeing the objects and events of the computer world makes that world much more tangible. Once accustomed to the SonicFinder, using quiet Finders is comparable to wearing earplugs in everyday life. Direct engagement as a quality is difficult to specify, and, I suspect, even more difficult to reveal through user testing. Experience with the SonicFinder, however, has convinced me that direct engagement is an important aspect of the user's experience.

The second major gain associated with the addition of auditory icons in the SonicFinder has to do with the increased flexibility it offers users. As has already been discussed, the use of sound to indicate when an object has been dragged over a container into which it might be dropped seems more useful than the visual feedback provided in this situation. The display of size information in several of the interactions may be useful in certain circumstances, for instance, in judging whether a file or group of files may be copied onto a nearly full disk. These are both examples where auditory feedback seems useful, whether or not it is redundant with graphic feedback.

The auditory icon that accompanies copying may illustrate this increase in flexibility more clearly (see Figure 5). When an object or group of objects is copied, the sound of pouring water accompanies the event. As the process continues, the frequency of the sound is continuously increased to indicate its progress by analogy with the sound a container makes as it is being filled. In current versions of the Finder, the progress of the copy operation is visually displayed using a "dial bar," a horizontal rectangle in which the proportion of filled area indicates the percentage of the copy that is complete. The copying sound thus presents information that is completely redundant with the visual feedback. But the graphic indicator requires the user to attend to the screen, whereas the sound does not. During lengthy copy operations (e.g., when a large number of files are copied), the advantage of using an auditory icon to display progress is obvious and pronounced.

Gains in direct engagement and flexibility make the SonicFinder an appealing interface. But such advantages appear difficult to demonstrate empirically, and no formal user testing has been done as of this writing. For the most part, increases in speed or accuracy associated with the addition of sound to this interface seem likely to be small or none. Instead, the auditory icons used in the SonicFinder are more likely to increase user satisfaction, which is difficult to measure, but obviously important. Finding situations in which sound can convey information better than graphics is more likely to lead to measurable influences. So, for instance, it might be that subjects would be faster in hitting a target when sound indicates the hit than when only visual feedback is used; they might also perform better in a dual-task experiment involving copying if the auditory icon indicating status is present. In addition, the SonicFinder might also be expected to be of value to visually disabled—although not blind—users (see Edwards, this issue). In any case, the SonicFinder demonstrates the functionality of sound both in terms of its ability to increase the feeling of working with a virtual world and in freeing users to listen to, as well as hear, the events in this world.

6. WHAT MAKES A DISPLAY INTUITIVELY ACCESSIBLE?

If the SonicFinder is successful, then it is because it increases the reality of the computer world and allows more flexibility to users. My contention is that these advantages are not merely due to the addition of sound to the interface, but rather to the particular strategy used in adding sounds. That is, I claim that the intuitive mappings between auditory icons and the events that they represent account for much of the increase in direct engagement, as well as the ease of obtaining the information offered by sounds. But this assertion raises several issues: What makes a mapping intuitively obvious? Auditory icons are based on literal-minded metaphors between events in the computer and those of the everyday world. This is valuable in that, being built on existing skills, their significance can be understood by the user with minimal training. However, will this sort of mapping limit the kinds of information that can be provided by auditory icons, and if so, how might these limitations be overcome?

In using sound to convey information from the computer, important qualities and quantities of the computer domain must be mapped to perceptible attributes of sound. I have suggested that basing this mapping on the ways everyday sounds correspond to attributes of their sources should make the mapping intuitively obvious both in its creation and interpretation, because such mappings exploit the causal structure in terms of which everyday listening tends to be organized. Here I discuss the role and nature of these sorts of mappings in the interface to explain why I think this is so.

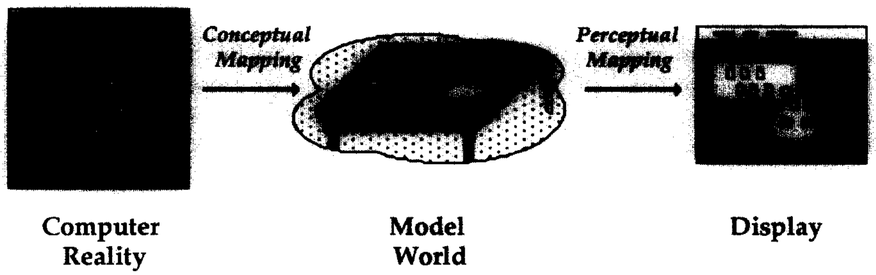
6.1. Conceptual and Perceptual Mappings

Our understanding of what computers do is built upon layers of metaphor (Hutchins, 1986). Most users, when concerned with performing some task, pay little attention to the computer as machine. Instead, they think of manipulating files, opening windows, accessing databases, and the like. All these tasks are based on a number of metaphors that lead from the domain of the computer hardware to that of the task. For instance, the configuration of a silicon gate at the hardware level might be seen as a bit of information at another level, as determining the value of a variable at a third, specifying a data structure at a fourth, or as giving rise to the behavior of a text file at an even higher level. These conceptual leaps from level to level are accomplished via metaphor, from electronics to information, information to variables, variables to structures, and structures to files.

The purpose of using metaphors in describing the operations of computers is twofold. First, metaphors allow everyday knowledge to be generalized to an unfamiliar domain. Speaking of computer entities as "files" obviates the need to know of data structures, much less the state of silicon gates. Second, a metaphor allows knowledge about a group of entities at one level to be summarized in terms of an entity at the next: so a group of variables form a data structure, a data structure with the memory it points to forms a file, and so on. Metaphors give organization and structure to groups of disparate entities at one level, integrating them so they may be thought of as familiar units at a higher level.

From this perspective, there are at least two mappings to be considered in any interface (see Figure 6). Most fundamental is a *conceptual mapping* between events in the computer and those in the model world in which some task is to be performed. The *computer reality* refers to the domain in which computer events are described, either by reference to the physical hardware of the system or its operations expressed in some programming language. When

Figure 6. **Two mappings exist between the reality of the computer and the display to which the user has access. The first is a conceptual mapping between events in the computer and those in some model world (e.g., that of a desktop). The second is a perceptual mapping between events in this model world and their perceptible manifestations, be they visual or auditory.**



metaphors are used to allow events in the computer reality to be understood in terms of a task domain, a *model world* is created. The conceptual mapping, then, is that which links corresponding aspects of the computer reality and the model world. The desktop metaphor, for instance, is the result of a conceptual mapping.

Perceptual mappings are made between the model world of the computer and its perceptible display. The conceptual mapping translates the reality of the computer into a conceptualization of a model world, say that of a desktop, but it does not make that world accessible to users. Instead, the objects and events of the desktop must be mapped further to their perceptible forms, be they visible, audible, tactile, or even olfactory. Files, for instance, might be made apparent by listing or speaking their names, by iconic representation, or by specific auditory icons. The choice of how entities and events in the model world are to appear is the result of a perceptual mapping.

The model world created by conceptual mappings is seldom a completely consistent one. Even in so-called desktop systems, the metaphor is usually mixed by the provision of menus, windows, and disks. This inconsistency may in part be because perceptual mappings are usually limited to the visual modality. Nonetheless, the model world by definition provides the most organization and integration with respect to the task domain. For this reason, perceptual mappings should be designed so they are consistent with this world. Conversely, the model world is seldom made explicit except through the displays that result from perceptual mappings. It is consistent to the degree that these displays indicate a consistent underlying metaphor. Systematic perceptual mappings that are consistent with a model world increase the organizational power of this world, whereas perceptual mappings that are incompatible with the model world can significantly disrupt its usefulness.

6.2. Types of Perceptual Mapping

The conceptual mappings that generate the model world of the computer are usually, perhaps always, metaphorical. Their purpose and power is to make the physical reality of the computer comprehensible in terms of some everyday task domain. Too literal or too arbitrary a mapping will produce a model world just as incomprehensible as the physical reality of the computer.

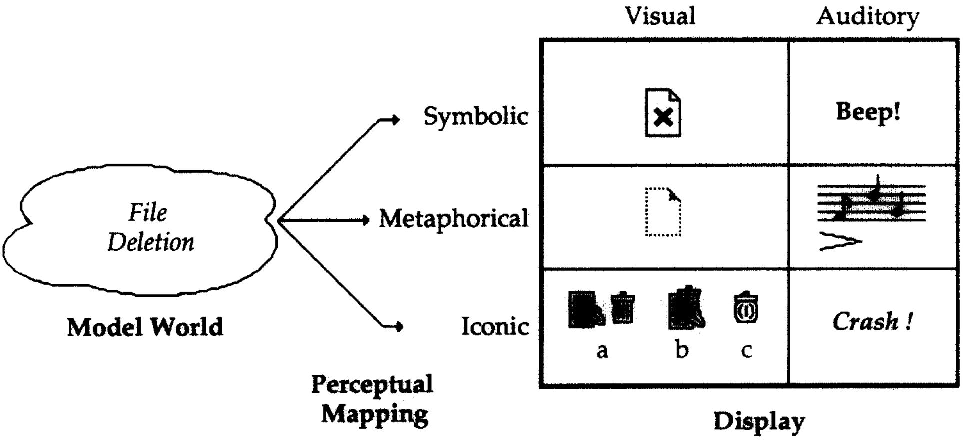
The nature of perceptual mappings, on the other hand, may vary widely. This is because perceptual mappings are not between the computer reality and the outward manifestations that are accessible to users, but between these manifestations and the model world of the computer. Perceptual mappings can range in how closely they reflect this model world. I (Gaver, 1986) distinguished three kinds of mapping (using an amalgamation of concepts from Bates, 1979; Heil, 1983; Peirce, 1932). According to this account, the relation between a display entity and the model world of the computer can be *symbolic, metaphorical,* or *iconic* ("nomic" in Gaver, 1986). A symbolic mapping is essentially arbitrary, depending on convention for its meaning. Metaphorical mappings make use of similarities between the entites or relationships of their domains, and thus are more constrained than symbolic mappings. Finally, iconic mappings are based on physical causation—the display entities look or sound like the things they represent.

This last kind of mapping, named after Peirce's (1932) use of the term, is that which gives rise to many sorts of visual icons (thus the name *icon*). An icon does not imply a literal, pictorial, or recorded mapping, instead its characteristics are causally related to the things it represents. So an icon may be a sketch, outline, or caricature; an auditory icon may be a recorded sound, or one synthesized to capture important features of an everyday sound. What is important is that the attributes of the representation convey information by virtue of their causal relations to the attributes they represent.

A given entity of the model world might be related to its display by any of these three mappings. For instance, Figure 7 shows alternative visual and auditory representations that indicate the deletion of a file. A symbolic visual mapping for deletion might be a file icon with a "x" in it. A corresponding auditory display might use a distinctive tone both when the file is deleted and if the user tries to access it afterwards. Both sorts of displays depend on the user's having learned the meanings of these arbitrary symbols for their effectiveness.

A metaphorical visual representation of deletion might use a faded file icon, relying on an analogy between deletion and disappearance. A corresponding auditory signal could involve a motif standing for the file which goes from loud to soft (see Blattner, Sumikawa, & Greenberg, this issue). An auditory signal such as this depends on an analogy between deletion and quieting or fading

***Figure 7.* A conceptual file deletion may be mapped to a display in many different ways. Here six possibilities are shown, one visual and one auditory example each of symbolic, metaphorical, and iconic mappings between the event and the display.**



into the distance. These kinds of metaphorical representations may effectively convey the notion of disappearance, but they are inconsistent with the desktop metaphor: Things on a desk do not simply fade away.

Finally, an iconic representation might involve dragging the file icon to a trashcan and throwing it in: Conceptual deletion is indicated by perceptual disappearance, and the visual representation of this event is causally related to the conceptual events in the model world. An iconically related auditory icon might be a crashing noise, indicating the destruction of the file.

Iconic mappings have several advantages over symbolic and metaphorical mappings. Iconically mapped displays are more closely related to the events in the model world they are meant to represent than either symbolic or metaphorically mapped representations. In Hutchins, Hollan, and Norman's (1986) terms, iconic mappings produce displays with a high degree of "articulatory directness": Their form echoes their function. Truly symbolic mappings are entirely unconstrained in terms of their form, whereas metaphorical mappings are only partially constrained. Iconically mapped displays are in principle wholly constrained because they are based on the laws relating physical events in the world to their perceptible manifestations. Insofar as users have knowledge about the relations between events, sights, and sounds in the everyday world, they may be expected to understand iconic mappings.

Iconic displays are based on the kinds of physical mappings that make the everyday world accessible to our senses, those that are described by physics. This implies not only that a single iconically mapped display will be interpretable, but that an entire group of displays, visual and auditory, will also be systematic. Visual and auditory icons can be related as visible and audible events are in the everyday world. That is illustrated in the examples of deletion displays shown in Figure 7: The iconic visual and auditory representations seem to fit together better than do either of the other pairs.

An intuitive mapping, then, is one that is constrained as much as possible by the kinds of correspondences found in the everyday world, and thus reflects the model world of the computer closely. Because auditory icons are based on the ways events in the everyday world make sounds, they are likely to involve iconic mappings to the events they portray. At least, auditory icons are likely to involve metaphorical mappings between events, as opposed to between an attribute of sound (e.g., loudness) and event. Because of this, their creation and interpretation are likely to be most constrained, both for a given auditory icon and for a system of auditory icons. Moreover, these constraints are usually the same as those that operate in the everyday world with which designers and users are most familiar.

6.3. Beyond Literal Mappings: Sound Effects and Source Metaphors

The constraints of using iconic mappings to create auditory icons are likely to help in making them intuitively appealing and systematically coherent. But such constraints also threaten to limit their application in computer interfaces. Where events in the model world correspond to sound-producing events in the everyday world, mapping information to auditory icons is not difficult. For instance, choosing a sound to accompany dragging is not problematic because dragging has an everyday world counterpart that makes sound. But what sound should accompany an event that does not exist in the everyday world—for instance, a disk-write error? What sound should be used to indicate an event whose everyday world counterpart does not produce sound, or where the sound it makes does not convey information relevant to the user? Iconic mappings between sonic events and computer events are desirable, as I have previously argued. But they are not always practical, because the computer is an artifact in which events do not always map neatly to events in the everyday world. If auditory icons are to be generally useful for computer interfaces, it must be possible to extend the mappings used to create them beyond the literal ones used for most of the auditory icons in the SonicFinder. Two ways of accomplishing this were found in developing the SonicFinder, both of which retain the intuitive mapping between display and event that characterizes auditory icons.

The first kind of problem—developing an auditory icon for an event with no counterpart in the everyday world—was encountered in finding a sound to accompany opening and closing windows in the SonicFinder. The sound of a real window opening and closing seemed inappropriate, as did sounds made by other related events. The problem is that windows in this interface are not like windows in the everyday world: They do not slide open, they "zoom" open, expanding from their associated icon. In the end, a "whoosing" sound seems much more appropriate to indicate opening windows.

This example suggests that in many cases, auditory icons should be more like movie sound effects than like typical everyday sounds. Instead of using naturally occurring sounds, new sounds will have to be developed that "sound like" events that occur only in the model world of the computer. Note that sound effects are not arbitrarily related to their associated events. Instead, they seem to rely on the abilities of listeners to generalize their knowledge about everyday sound-producing events to new ones, even imaginary ones involving things such as light-sabers or transporters. Windows in the everyday world don't open as the ones in the SonicFinder do, but this event does resemble others in the everyday world, such as the rapid approach or sudden expansion of an object. Sounds made by these kinds of events are thus more appropriate as auditory icons for opening windows in the SonicFinder. Examining the ways sound effects are created seems to have great potential for extending auditory icons to situations in which literal analogies between computer events and events in the everyday world are impossible to find.

The second kind of problem — finding a sound to accompany an event that does not make an informative sound in the everyday world — was encountered in creating an auditory icon to represent copying. Although xerographic reproduction might be used as an everyday-world counterpart to copying in the computer, using the sound of a copier seemed inadequate in this situation. First, copying in the computer is a more continuous process than using a copier, in which separate pages are copied in separate stages. Second, the sounds made by copiers in the everyday world do not change to indicate the amount completed, and this information is highly relevant to users (Myers, 1985). So, because the analogy between these two forms of copying seemed inadequate, and because the everyday sound did not convey information felt to be relevant for the computer counterpart, a different sound-producing event was required for a useful auditory icon. In current versions of the SonicFinder, the sound of liquid pouring into a container is used to accompany copying. As pointed out earlier, such a sound is useful in that information about nearness to completion is highly salient.

The pouring sound used to indicate copying is not a sound effect nor is it arbitrarily related to the computer event about which it provides information. Moreover, using pouring as an analogy for copying does not depend on a metaphorical mapping between sound and event. Instead, this auditory icon relies on an analogy between copying in the model world and pouring in the everyday world. The sound itself is iconically related to its source, but this auditory icon is metaphorical in terms of the mapping between events. In fact, using a pouring sound to represent copying might perhaps change the conceptual mapping between the event in the computer reality and that in the model world. This is particularly plausible given that the dialbar used to indicate process is essentially a graph-like status indicator. There is no ongoing visual reinforcement of a particular model world representation of copying—although of course the result is that a copy is made. In any case, basing auditory icons on metaphors between events in the model world and others in the everyday world is another way to extend the range of auditory icons.

Creating sound effects or developing metaphorical mappings between events in the model world and sound-producing events in the everyday world are two ways to overcome the limitations of auditory icons. Both are promising in that they seem to loosen the constraints implied by relying on a literal mapping between sounds and events in the model world, while retaining the intuitive accessibility of auditory icons. Sound effects may be developed for events that do not exist in the everyday world, and well-designed effects will still be based on causal relations to these events—that is, they will be iconic. When metaphorical auditory icons are used, they rely on analogies between sound-producing events, instead of analogies between events and attributes of sound. The perceptual mappings involved in such auditory icons are iconic, and the conceptual mapping may itself change to reflect the auditory event. Because they retain the intuitive accessibility that seems to accompany iconic mappings, these methods for extending auditory icons seems more promising than resorting to more arbitrary systems of using sound to convey information.

7. FUTURE DIRECTIONS

The SonicFinder demonstrates a number of the attractive characteristics of auditory icons. The sounds seem to be naturally integrated into the interface and appear intuitively accessible. They increase feelings of direct manipulation and provide flexibility in interacting with the model world of the computer. In some cases, they provide information that otherwise must be requested, and in others, they seem more appropriate than do graphical displays of identical information. Finally, auditory icons seem to add significantly to the satisfaction of using the interface.

These advantages are important, but several potential benefits of auditory icons are not realized in the SonicFinder. Most notably, sounds in this interface convey little relevant information that is not redundant with that displayed visually. This is because the auditory icons used in the SonicFinder reflect the model world that had already been developed for the Finder. In the long run, auditory icons may be expected to influence the conceptual mapping that produces this model world. One of the most attractive possibilities for the use of auditory icons is in reducing the clutter of visual displays, particularly in multitasking systems. In the future, sound might be used to indicate user accessibility of files or programs, their age or frequency of use, or the amount of memory associated with their use. Users might choose whether certain types of information (e.g., progress indicators) should be displayed graphically or via auditory icons. Sound might also provide information about the status of background processes, the number of links in a networked environment, and other factors of the computer environment.

One important application for sound, as Edwards (this issue) pointed out, is in modifying graphical interfaces so that they are accessible to the visually impaired (a problem that has more general implications, e.g., in using such interfaces over the telephone). The perspective on everyday listening I suggested here may be useful in addressing this problem. Consider, for instance, the issue of finding ones way around the screen if it is not visible. This is analogous to navigating in the everyday world if one is blind. The problem here should probably not be formulated as one of knowing one's position in Euclidean space, but instead as one of knowing how to get to objects or locations of interest. A number of auditory cues seems to be useful to visually disabled people in accomplishing this task. Perhaps most important are what Jenkins (1985) called "auditory landmarks." These include relatively continuous sounds such as those of busy streets or ventilator fans as well as environments that modify sounds in reliable ways, such as echoing hallways or enclosed areas.

Such cues might be used in the interface in several ways. For instance, the cursor might be treated as a microphone which can sample sounds made by objects in the interface or by a number of single-purposed auditory landmarks placed by the user. In addition, the amount and kind of reverberation applied to any auditory icon might be used to indicate parameters of the environment that would be useful in navigation, such as the size of the local environment or the proximity to a boundary. Such approaches, based on the ways visually impaired people navigate in their daily lives, may be useful in providing spatial cues about the model world that are not only visual.

Applications that realize the full potential of auditory icons will certainly depend on a careful analysis of the interface as well as a good understanding of everyday listening. They are also likely to require the ability to make fairly complex manipulations of sound and an imaginative use of sound effects and source metaphors. The SonicFinder seems a promising start towards demonstrating the potential of auditory icons, but however satisfying it may be, it is only a beginning.

Acknowledgments. Many thanks go to Sara Bly, Bill Buxton, Don Norman, and Anne Schlottmann for their advice and encouragement. I also thank John Meier, Mark Lentzcner, Jon Worthington, Joy Mountford, and the Human Interface Group of Apple Computer, Inc. for their help and support in creating the SonicFinder.

Support. This research was supported by a grant to Donald A. Norman and David E. Rumelhart from the System Development Foundation. Support was also provided by funds from the Affiliates of Cognitive Science of UCSD and by contract N00014-79-C-0323, NR 667-437 to Donald A. Norman from the Personnel and Training Research Programs of the Office of Naval Research. The SonicFinder was developed while I was an employee of Apple Computer, Inc.

REFERENCES

Apple Computer, Inc. (1988). *Inside Mac* (Vol. 5). Menlo Park, CA: Addison Wesley.

Ballas, J. A., Dick, K. A., & Groshek, M. R. (1987). Failure to identify "identifiable" sounds. *Proceedings of the 31st annual meeting of the Human Factors Society.* Santa Monica: Human Factors Society.

Ballas, J. A., & Howard, J. H., Jr. (1987). Interpreting the language of environmental sounds. *Environment and Behavior, 19,* 91-114.

Ballas, J. A., & Sliwinski, M. J. (1986). *Causal uncertainty in the identification of environmental sounds* (Tech. Rep. No. ONR-86-1). Washington, DC: Georgetown University, Department of Psychology.

Bates, E. (1979). *The emergence of symbols: Cognition and communication in infancy.* New York: Academic.

Bly, S. (1982). *Sound and computer information presentation* (UCRL-53282). Unpublished doctoral dissertation, Lawrence Livermore National Laboratory and University of California, Davis, CA.

Buxton, W. (1986). There's more to interaction than meets the eye: Some issues in manual input. In D. A. Norman & S. W. Draper (Eds.), *User centered system design: New perspectives on human-computer interface* (pp. 319-337). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Freed, D. J., & Martins, W. L. (1986). Deriving psychophysical relations for timbre. *Proceedings of the International Computer Music Conference* (pp. 393-405). The Hague, The Netherlands: Computer Music Association.

Gaver, W. W. (1986). Auditory icons: Using sound in computer interfaces. *Human-Computer Interaction, 2,* 167-177.

Gaver, W. W. (1988). *Everyday listening and auditory icons.* Unpublished doctoral dissertation, University of California, San Diego, CA.

Heil, J. (1983). *Perception and cognition.* Berkeley, CA: University of California Press.

Hutchins, E. L. (1986). *Metaphors for interface design.* Paper presented at NATO Workshop on Multimodal Dialogues Including Voice, Venaco, Corsica, France.

Hutchins, E. L., Hollan, J. D., & Norman, D. A. (1986). Direct manipulation interfaces. In D. A. Norman & S. W. Draper (Eds.), *User centered system design: New perspectives on human-computer interaction* (pp. 87-124). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Jenkins, J. J. (1985). Acoustic information for objects, places, and events. In W. H. Warren & R. E. Shaw (Eds.), *Persistence and change: Proceedings of the first international conference on event perception.* Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Laurel, B. (1986). Interface as mimesis. In D. A. Norman & S. W. Draper (Eds.), *User centered system design: New perspectives on human-computer interface* (pp. 67-85). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Loy, G. (1985). Musicians make a standard: The MIDI phenomenon. *Computer Music Journal, 9*(4), 8-26.

Mansur, D. L., Blattner, M. M., & Joy, K. I. (1985). Sound-graphs: A numerical data analysis method for the blind. *Proceedings of the 18th Hawaii International Conference on System Sciences, 18,* 163-174.

Mezrich, J. J., Frysinger, S., & Slivjanovski, R. (1984). Dynamic representation of multivariate time series data. *Journal of the American Statistical Association, 79,* 34-40.

Myers, B. A. (1985). The importance of percent-done indicators for computer-human interfaces. *Proceedings of the CHI '85 Conference on Human Factors in Computer Systems,* 11-17. New York: ACM.

Peirce, C. S. (1932). In C. Jartshorne & P. Weiss (Eds.), *Collected Papers of Charles Sanders Peirce.* Cambridge, MA: Harvard University Press.

Vanderveer, N. J. (1979). Ecological acoustics: Human perception of environmental sounds. *Dissertation Abstracts International,* 40/09B, 4543. (University Microfilms No. 8004002)

Warren, W. H., & Verbrugge, R. R. (1984). Auditory perception of breaking and bouncing events: A case study in ecological acoustics. *Journal of Experimental Psychology: Human Perception and Performance, 10,* 704-712.

HCI Editorial Record. First manuscript received June 13, 1988. Revision received November 2, 1988. Accepted by William Buxton. —*Editor*