

Designing Auditory Cues for Mercator

Elizabeth D. Mynatt

Xerox PARC
mynatt@parc.xerox.com

ABSTRACT. The practice of designing human-computer interfaces requires synthesizing knowledge about the intended users, their tasks and interface design guidelines. This task is even more challenging for designers of auditory interfaces since few useful guidelines exist. To provide some general intuition about how people identify and conceptualize various auditory cues, we conducted two experiments. The purpose of this paper is to discuss how we used the results of these experiments to guide the selection of sounds for Mercator. As a screen reader for blind users, Mercator transforms graphical interfaces into auditory interfaces. The selection of the specific auditory cues used to convey objects and events in the Mercator interface is the result of this work. This design process was not solely the consumer of these experimental results, but also produced further insight into the selection and use of auditory cues.

KEYWORDS: Auditory Interfaces, Auditory Icons, Design Methodology, Audio, User Interfaces

1. INTRODUCTION

The practice of designing human-computer interfaces requires synthesizing knowledge about the intended users, their tasks and interface design guidelines. These guidelines, abstracted from the previous experiences of numerous designers, often provide ideas as well as constraints, but they rarely point to one, unquestionable answer. Additionally designers often face practical constraints such as a limited palette of colors or restricted screen space. Thus, the designer must work within this space of users, tasks, guidelines and practical constraints to carve an artifact balanced across these dimensions.

This task is even more challenging for designers of auditory interfaces. Few useful guidelines exist. Most interface guidelines simply advise not using audio at all. Acknowledging that audio is often useful in many human-computer tasks, designers still struggle with deciding what kinds of sounds to use, not to mention which specific sounds to use.

This paper reports on the process of deciding which sounds to use in an interface design. The design context is a system called Mercator. Mercator, described in more detail in the following section, transforms graphical interfaces into nonvisual interfaces for blind users. One class of sounds, called auditory icons, was particularly attractive for this design task since auditory icons operate much like graphical icons in that they leverage knowledge of the everyday world.

To provide some general intuition about how people identify and conceptualize various auditory cues, we conducted two experiments. In the first experiment, we analyzed descriptions of sounds. In the second experiment, we asked subjects to match sounds to concepts in graphical interfaces to understand how these aspects of graphical interfaces could be translated into sounds. In this paper, we briefly summarize these experiments and describe observations from the data analysis.

The purpose of this paper is to discuss how we used the results of these experiments to guide the selection of sounds for Mercator. Like other guidelines in human-computer interface design, the experimental results rarely pointed to one answer, in this case, a particular sound. More often they offered multiple hints, sometimes contradictory, during the selection process. This design process was not solely the consumer of these experimental results, but it also produced further insight into the selection and use of auditory icons.

2. BACKGROUND

2.1. Auditory Icons

Gaver introduced the concept of auditory icons based on the premise that people describe sounds relative to the objects interacting to make the sounds. Computers and other man-made devices make a variety of beeps, buzzes and other artificial noises that otherwise we would never hear. Conversely, we typically hear things crumbling, things sloshing, things colliding - we hear the results of objects interacting together. When asked to describe a sound, we tend to describe it in terms of the objects that generated the sound such as a door slamming, stairs creaking or glass breaking. In other words, we describe sounds in terms of their sources, not in classical terms such as pitch and duration.

What this realization means to an interface designer is that we can use sounds to make users think of familiar objects in the same way as icons in GUIs. Gaver first demonstrated the use of auditory icons with his interface SonicFinder. In SonicFinder, Gaver added auditory cues to the graphical Finder environment on the Macintosh. Dragging a file icon with the mouse sounded like dragging something along the floor. Dropping an icon into the trashcan sounded like dropping a large object into a metal trashcan.

Auditory icons are not limited to simply reflecting categories of events and objects, but can be *parameterized* to reflect their relevant dimensions as well. For example, the auditory icon for a file can be manipulated to convey the size of the file. Although Gaver has successfully parameterized auditory icons in a number of interfaces, he notes the difficulty in parameterizing auditory icons due to the difficulty in controlling sampled sounds along the relevant dimensions and suggests synthesis as an alternative to sampled sounds.

Auditory icons appear to be a good match for this design task. Since they mimic the use of graphical icons, they seem well suited to conveying the components of graphical interfaces. Likewise, since their design potentially leverages existing knowledge, their use may help minimize the blind user's cognitive load.

2.2. Mercator Interfaces

Mercator is a system that automatically generates nonvisual interfaces for graphical applications. It was primarily designed to provide access to graphical interfaces for people who are blind. As a graphical application is executing, Mercator creates an off-screen

model of the application interface. From that model, Mercator generates a nonvisual interface to the graphical application. As users interact with the nonvisual interfaces, Mercator translates their input to the input expected by the graphical application; translating keyboard input to mouse input for example. Thus while a graphical application is executing, Mercator users are able to interact with a parallel nonvisual interface.

Users navigate Mercator interfaces by traversing the structure of the application interface. This hierarchical structure is based on the objects used to construct the graphical interface. Moving up or down accesses different levels (parent/child) of the hierarchy while moving left or right accesses sibling objects. Users cannot move out of the bounds of the interface structure. Keyboard shortcuts allow users to jump to different portions of the structure and switch between windows on the screen. Separate controls are provided for manipulating the interface. This separation of navigation and control allows the user to navigate the interface without fear of accidentally causing an event to occur.

Auditory icons convey the types of objects in the interface as well as provide feedback for interface events. For example, when the user navigates to a text-entry area they hear the sound of a typewriter. These auditory icons are parameterized to convey attributes of the objects. For example, a grayed-out toggle button sounds like a muffled pull-chain light switch.

Synthesized speech is used to read the text in the interface, such as the label on a button, as well as provide context-sensitive help. To access controls for scanning and editing large text areas, users switch Mercator into *text mode*. While in text mode, the keyboard controls are remapped to text controls.

3. USING AUDITORY ICONS

Evaluations of auditory interfaces reveal a number of factors that affect the usability of auditory icons:

- **Identifiability** - The user must be able to recognize the sound. The ecological frequency (how common the sound is) versus the relative uniqueness of the sound help determine its overall identifiability. If the user is unable to correctly identify the sound, then the usefulness of the sound is likely to be reduced. For example, in Cohen's ShareMon interface, the sound of a key turning in a lock represented a registered user logging into a computer system. Although the mapping between the sound and the interface concept is intuitive, the sound was not identifiable and therefore not utilized.

- **Conceptual Mapping** - How well does the sound map to the aspect of the user interface it is representing? Sound seems to lend itself to storytelling. Evaluators of auditory interfaces are often amazed at the stories users will build to explain the sounds that they hear. A favorite example is again taken from the ShareMon system where a knocking sound immediately followed by the sound of footsteps indicated that a user had logged onto a computer system and was accessing the shared file space. One test subject explained the sounds by describing a father expecting a baby who was knocking at a door (no one responds) and then pacing back and forth. The metaphorical power of auditory icons is immense, but controlling these mappings is a difficult design task.

- **Physical Parameters** - The physical parameters of the sound such as length, intensity, sound quality (for example, sampling rate) and frequency range may affect its usability. These physical parameters not only affect the user's reaction to the auditory cue, but they also affect the usability of the cue from the designer's perspective. The overall quality of the sound is largely determined by the sampling process. Users may negatively react to cheap (phone quality) or noisy sounds. The frequency composition of the cue may affect how it can be manipulated with standard filters.

- **User Preference** - How the user responds emotionally to the auditory icon is also important. Is the sound harsh or too cute? Almost all evaluations of auditory interfaces uncover sounds that users do not like.

The auditory icons used in an interface must also be evaluated as a cohesive set, again addressing the above issues. Concerning identifiability, the cues must be relatively distinct from each other so that they can be individually identified. The conceptual mappings in the interface must not be counterintuitive. The physical parameters of the auditory icons such as length, intensity and sound quality must also be carefully controlled as the user might attempt to attribute some meaning to any perceivable differences. These issues are important even if the sounds are not simultaneously presented. In addition to the above issues, concurrent presentation of auditory cues must also address basic psychoacoustic issues such as masking and streaming.

4. IDENTIFYING AUDITORY ICONS

To assess the identifiability of common auditory cues, we collected 64 sounds (short "everyday" samples from ten sound effects CDs with minimal editing). During the experiment, the sounds were played over two

speakers to a classroom of 83 students. Under tight time constraints, subjects were asked to describe (free-form) the sounds as best they could.

The subjects' responses were analyzed by transforming their answer into a point in a 4x4 matrix. One axis on the matrix indicates how well the subject identified the object associated with the sound. The other axis indicates how well the subject identified the action associated with the sound. Per object and action, four scores were possible.

Y	Correct identification
P	(Partial) Either the subject identified a base material, an object with the same affordance or a similar action.
A	(Alternate) The subjects commonly selected a consistent (alternative), yet incorrect answer.
N	Incorrect or no answer

Table 1: Possible Identification Scores

4.1. Observations

Identifiability Scores are Low

The first result to note is that, in general, the identifiability scores were low with few sounds (13) showing an overall score over 80 (eighty) percent. This result is not surprising given the difficulty of the task and the short length of the sounds (under two seconds).

Distinction between Actions and Objects

The distinction between identifying a sound as an object or an action is clearly evidenced. Many sounds are consistently identified as objects (camera, typewriter, door, zipper) while other sounds are consistently identified as actions (closing, tearing, winding, locking). For example, for the sound of a camera taking a picture, specifically the shutter operating, subjects easily identified the object as a camera, but were unable to identify the correct action (taking picture, flashing, winding film). Another sound of winding a parking meter was easily identified as winding but the object (parking meter, typewriter, music box) was not identified. This separation supports the guideline for choosing some sounds to represent objects while using other sounds to represent actions. In general, the designer should decide if an action or an object (or both) need to be conveyed, and then choose the appropriate auditory cue.

Partial / Alternative Answers Merit Research

Many of the sounds resulted in high partial (P) or alternative (A) scores. In other words, the subjects heard some of the information in the sound. A partial answer for the object was recorded if the subject identified the base material of the object, or identified an object with the same affordance. For example, for the sound of closing a metal mailbox, recognizing the metallic material or answering "closing a file cabinet" resulted in a partial object score. For actions, identifying similar actions resulted in partial scores such as dropping or throwing, swinging or striking, or closing, latching or locking.

An alternative answer was scored if the pool of subjects tended to identify the object incorrectly but consistently. Cultural biases may partially explain alternative answers. For example, in our pool of undergraduate Georgia Tech students, whip cracks were identified as gun shots and popping a cork out of a champagne bottle was identified as making a popping noise with one's finger and cheek.

The large number of sounds with partial and alternative scores motivates future work in the learnability of auditory cues. One hypothesis is that sounds with high partial answers are easy to learn whereas sounds with high alternative answers are difficult to learn since it would require the subject to break the existing mapping between the sound and the alternative concept.

Sounds Naturally Conflict

Many sounds naturally conflict. For examples, machines that afford typing such as typewriters, keyboards, cash registers are naturally confused. Likewise printing and copying sounds are often mistaken for each other. These conflicts indicate when it may be difficult to present two similar concepts in an interface using auditory cues. For example, although the sounds of a copier and printer may be quite distinct, it may be difficult to correctly identify them when they are both used in the same interface.

5. MAPPING AUDIO ICONS TO GUIs

This experiment assessed how users expected auditory cues to be used to represent concepts common in graphical interfaces. Twenty-eight sounds were selected from the initial collection of sounds used in the first experiment. The sounds selected had high identifiability scores from the first experiment. Fifteen concepts (objects and actions) from graphical interfaces

were chosen as the independent variable. During the experiment, each subject was presented with the graphical interface concept and five sounds. Their task was to choose which sound best matched the interface concept.

5.1. Observations

The observations reported here are based on debriefing the subjects and a quantitative analysis of the numerical data gathered during the experiment.

Four Types of Selection

During the debriefing, the subjects described why they had chosen a particular sound for an interface concept. The reasonings used to guide their selections seem to fall into four major categories. First, the subjects (as predicted) chose sounds that they felt represented the *semantics of the interface concept*. For example, the sound of typing on a typewriter (multiple keystrokes) was a common choice to represent an editable text field. Likewise, for closing a window, the sound for closing a car door was a common choice, followed by the sound of closing a zipper. This observation is not surprising, but it does indicate that sounds can be used to represent semantic information in the interface.

Second, several subjects indicated that they choose sounds that mapped well to the perceived *importance of the interface concept*. For example, the sound of a short tap on a typewriter was the most common sound for the check boxes that implement nonexclusive choices. Subjects reasoned that selecting a check box is easily undone, and therefore less important than pushing a button that was represented by a longer, more complex sound such as cracking a whip.

Third, subjects also indicated that they chose sounds that mapped to the perceived *length of the interface concept*. For example, selecting a radio button (common sounds: short and long pops) was described as taking less time than opening a folder (common sounds: opening door and slide whistle-pitch ascending).

Fourth, the subject choose sounds that mapped to the *physical appearance of the interface concept*. For example, a slide whistle (pitch descending) was the most common sound for a pull-down menu. Subjects explained that pop-up menus would have a different sound although the menu concept was shared by both objects. Several subjects explained that the sound of a camera shutter mapped well to the concept of a window appearing on the screen since the sound reminded them of square shapes being presented in quick succession.

Feedback is the Expected Use of Sound

Subjects' comments during the debriefing indicated that they expected the auditory cues to be used as feedback in the interface, usually as the result of a direct manipulation task. This use is in contrast to using sound to simply convey the identity of interface objects independent of a particular action. For example, in Mercator, navigating to an object causes a sound to be played that conveys the type of object. This distinction between conveying the identity of an object separate from feedback resulting in manipulating the object is important in auditory-only interfaces where the contents of the interface must be conveyed aurally.

6. UTILIZING RESULTS IN MERCATOR

The primary purpose of running the identification and mapping experiments was to gain insight into how people described auditory cues, and how they expected those cues to be used in computer interfaces. A secondary purpose was to select cues to be used in Mercator. The collection of sounds used in the experiments were samples taken from over ten sound effects CDs. When copying sounds from the CDs, we chose sounds that were good candidates for use in Mercator. For example, we looked for short, generic sounds for actions such as closing, opening, latching, winding and so on. We also looked for sounds involving objects such as typewriters, doors, file cabinets. We avoided two major types of sounds, animal sounds, and ambience sounds. For the most part, the experiments did not result in clear choices for which sounds to use in Mercator, but offered guidelines for the selection. This selection process is described in the following sections. In most cases, we attended to multiple guidelines when choosing sounds for Mercator. Table 2 illustrates the guidelines that we were able to use in the design.

6.1. Isolating Optimal Auditory Icons

Prior to performing the experiments, we hypothesized that a typewriter sound should represent text areas while a printer sound should represent read-only text areas such as message bars. But the question of which typewriter and printer sounds to use remained. The initial identifiability experiment helped isolate sounds that were readily identified as typewriters and printers. In the case of the typewriter sound, the mapping experiment confirmed that a typewriter sound would work. The mapping experiment did not distinguish that a printer sound was a good match for a message area.

However, debriefing test subjects indicated that some subjects interpreted the message bar as only being used for conveying errors. These subjects choose alarm sounds instead. In general, the current printer sound is not ideal and we are still looking for a better one.

6.2. Action- & Object-oriented Sounds

One surprising result of the identifiability experiment was that some sounds were more often described as actions, while some sounds were more often described as objects. This bias in identification is useful in Mercator where some sounds need to represent objects independent of an action. When choosing sounds for objects, we looked for an indication of this bias. For example, we tried to identify push button sounds where the user did NOT think that they had just pressed a push button.

The opening door sound for containers is still problematic. Novice users typically believe that they have *entered* a container, not that they can look into a container. Previous sounds of rattling lids or shaking boxes with objects in them were not readily identified.

To represent selection, we needed a sound that had a strong action bias that did not remind the user of any specific object. The ripping paper sound has been successful. Most users do not identify the object as paper. Some do not recognize the action as ripping, but all users seem to believe that they have caused something to occur.

Obviously how users interpret sounds is also dependent on their input to the system. If users press the Enter key, they should anticipate that a selection is occurring. But selecting an object is generally followed by a series of sounds to represent actions and new objects appearing on the screen. Users have never identified the selection sound as a new object.

6.3. Controlling the Length and Complexity

Based on their selections and comments during the mapping experiment, subjects indicated that the length and complexity of an auditory cue should be consistent with the concept that it represents. Although not fully defined, complexity seems to be based on the range of pitch conveyed in a sound and the transitions between those pitches. Check boxes appear to be the simplest interface concept so they are conveyed with a short sound somewhere between a click and a keypress. As push buttons appear to be more complex since selecting a push button is harder to undo, this sound is longer and more complex.

The whistling sounds were popular choices since these sounds are particularly engaging. We use these sounds to represent pop-up windows since these sounds are relatively long and complex, matching the relatively lengthy and important occurrence of a pop-up window.

In a similar manner, a relatively long winding sound is used to indicate when an application is connecting to Mercator. Winding seems to imply waiting for an indeterminate period of time. In contrast, disconnecting an application is a short process and is accompanied by a succinct flushing sound.

6.4. Mapping Cues to Visual Events

Although some aspects of the visual interface are not directly conveyed to the Mercator user, representing visual events with sounds is sometimes useful. As previously discussed, whistling sounds indicate the appearance and disappearance of a pop-up window. As the pitch rises, it sounds like something is coming toward the user, to the forefront. This experience is like a pop-up window appearing on top of other windows. Although other sounds could indicate the presence of something new, the whistling sound helps solidify the concept of an appearing pop-up window. A pop-up window (the object) is conveyed with the sound of a spring for the same reason. We are still looking for a good “pulling down” sound to represent a pull-down menu.

Windows are often described as pieces of paper in the desktop metaphor. For this reason, switching between windows is accompanied with the sound of shuffling through paper. Again, the sound conveys a change in stacking order on the screen. Although the windows are not visible to the Mercator user, it is helpful to reinforce common concepts in graphical interfaces. In this way, Mercator users can more knowledgeably talk to their sighted colleagues about the behavior of a graphical application.

6.5. More Abstract Representations

Although our initial designs focused on concrete representations, some concepts in the interface did not map well to concrete auditory icons. The most difficult concept to find a sound for was entering and exiting text mode. Concrete representations such as the sound of turning on a typewriter, or winding paper in the typewriter were more confusing than helpful. Since switching into text mode is caused by the user hitting a reserved key, the user only needed a simple feedback sound confirming the action. We chose the sound of a drawer being pulled out of a chest of drawers. The sound

is simple, short and its low pitch helps it to be non-intrusive. The rolling and rocking sound indicates a change of state that does not sound alarming.

Radio buttons and toggle buttons were difficult to represent because in both cases, we wanted a simple, two-stage sound. The two parts of the toggle button sound needed to indicate that something could be latched, or unlatched. The two parts of the radio button sound needed to indicate that selecting one button might result in another button being deselected. We used a pull chain light switch sound for the toggle button. This concrete representation should remind the user of a control that can be turned on and off. For the radio button, we used a more abstract push button sound where the first part is shorter and lighter, indicated the deselection, than the second part. We also looked for a cue that sounded *round* since radio buttons are usually round as opposed to other push button controls¹. Indications of roundness included no sharp attacks or decays and a simple, harmonic structure.

For the navigation error sound, we needed an action-oriented sound, but it needed to convey an attempted action. We wanted the sound to be non-alarming, no harsh buzzers or beeps, and to reinforce the concept of the hierarchical navigation scheme. We decided on a sound akin to a ball rebounding against a wall. It is a highly, action-oriented sound that hopefully conveys the walls of the explicit interface structure. When played successively (if the user hits too many keys in one direction), it creates a nice drumming effect. Although its low pitch is relatively non-intrusive, one frustrated test user working with Mercator described the sound as “banging his head against the wall!”

6.6. Designing Sets of Auditory Cues

The identification and mapping experiments focused on the design of individual auditory cues. These experiments were useful in that they highlighted general design guidelines. The net result, however, was that we used these guidelines to design a *set* of auditory icons for Mercator. Like many interface design tasks, it is difficult to design components of the interface separate from the context of the interface. One reason this statement is true for designing auditory icons is that the icons must be designed as a cohesive set. For example, the sound for flipping papers should not be louder than the sound for

1. In graphical interfaces radio buttons are generally round. In contrast, real world radio buttons are generally rectangular. In this design decision, the conventions of graphical interfaces overrode real world experience.

opening a door. The design guidelines for controlling the length and “complexity” of sounds are useful when comparing relative differences between sounds.

Another difficulty is choosing sounds for similar concepts in the user interface. For example, we needed sounds for push buttons, menu buttons, radio buttons, toggle buttons, and check boxes. These controls are all pushed (selected) to trigger an action in the interface. In Mercator, we used a combination of concrete and abstract sounds attempting to control the length and complexity of the individual cues. There is little chance that a successful set of icons would result from designing the auditory icons independently.

6.7. Practical Considerations

Mercator’s interface design also placed two practical constraints on the selection of sounds.

- **Interruption** - When navigating an interface, the user may opt to move quickly through the interface without waiting to hear the full length of the auditory cues. Although the sounds are short, a user with any reasonable level of expertise tended to move faster than the sounds could be played. To accommodate faster navigation, sounds needed to be identifiable in the shortest time possible. The initial pitch and complexity of the sounds seemed to influence their identification.
- **Filtering** - Multiple levels of information are often conveyed in one auditory cue such as representing a grayed-out push button or a relatively small text area. A common technique to add an additional layer of information is to filter the sound with a high- or low-pass filter. These filters dampen a range of frequencies present in the sound resulting in a cue that sounds muffled or excited. These filters combined with modifying the overall intensity of the sound are both fairly powerful and cheap to implement. The use of these filters, however, constrained the initial selection of sounds for Mercator. For a filter to affect a sound, it must have an appropriate range of frequencies. In short, all of the sounds used in Mercator have a reasonably large bandwidth of frequencies so that filtering can be used.

7. A NEW PERSPECTIVE

When we began designing Mercator’s auditory interfaces, our view of using auditory icons was limited to using concrete representations to represent concepts in the interface. The typewriter and printer auditory icons are examples of concrete representations where we want the user to identify the auditory icon. By thinking of a typewriter, the user is reminded that the object is an

editable text area. As we designed cues for more abstract concepts such as switching text modes, we began paying more attention to the metaphorical qualities of everyday sounds. Realizing that auditory icons did not have to utilize concrete representations, we began experimenting with sounds where we did NOT want the user to be able to identify the source of the sound, but we still wanted the user to apply everyday listening skills when interpreting the sound. This strategy is not novel as foley artists rarely use concrete sounds when designing sound effects for films.

This realization made us question recent work where designers compared real world and musical sounds. The study concluded that musical sounds should be used because “users consistently preferred musical sounds to real world sounds.” This result is not surprising since a professional designer created the musical sounds, but did not design the real world sounds. What the researchers did not acknowledge was that many of their “musical sounds” had significant real world qualities to them. For example, the musical sound for terminating a phone call ended abruptly and sounded like a metaphorical representation of placing the phone handset back into the cradle. Likewise, the musical sound for ringing was repetitive with a tempo akin to a real world phone ringing.

An interesting goal for future work is understanding what are the attributes or affordances of everyday sounds that help people leverage their everyday listening skills. In the previous examples, the amplitude envelope, pattern and tempo helped one to interpret the musical sounds in terms of everyday listening. The partial identification scores in the identification experiment may point to some of these underlying attributes.

8. REFERENCES

- Albers, M. The Varese System, Hybrid Auditory Interface, and Satellite-Ground Control: Using Auditory Icons and Sonification in a Complex, Supervisory Control System. In *Proceedings of the Second International Conference of Auditory Display, ICAD '94*, Sante Fe, New Mexico, 1995, pp. 3-14.
- Ballas, J. A. Delivery of Information Through Sound, In *Auditory Display: Sonification, Audification and Auditory Interfaces*, edited by G. Kramer, SFI Studies in the Sciences of Complexity Proc. Vol. XVIII, Addison-Wesley, 1994, pp. 79-94.
- Butler, D. *The Musician’s Guide to Perception and Cognition*. Schirmer Books. 1992.

Concept	Sound	Concrete or Abstract Representation	Action or Object	Control length and complexity	Map to Visual Events
Editable text area	Typewriter, multiple keystrokes	Concrete			
Read-only text area	Printer printing out a line	Concrete			
Push button	Keypress (ca-chunk)	Abstract		X	
Toggle button	Pull chain light switch	Concrete		X	
Radio button	Pa pop sound	Abstract		X	X
Check box	One pop sound	Abstract		X	
Window	Tapping on glass (two taps)	Concrete			
Container	Opening a door	Concrete	Object???		
Pop-up dialog	Spring compressed then extended	Concrete	Object	X	X
Application	Musical sound	Abstract			
Selection	Ripping papers	Abstract	Action		
Switching between applications	Paper shuffling	Concrete	Action		
Navigation error	Ball rebounding against wall	Abstract	Action		
Entering text mode	Rolling / rocking sound (drawer pulled out)	Abstract	Action		
Moving edit cursor in text area	Click (pitch based on position in text)	Abstract			X
Popup appearing / disappearing	Whistle up/down	Abstract	Action	X	X

TABLE 2: Choosing Sounds for Mercator

Cohen, J. Monitoring Background Activities, In *Auditory Display: Sonification, Audification and Auditory Interfaces*, edited by G. Kramer, SFI Studies in the Sciences of Complexity Proc. Vol. XVIII, Addison-Wesley, 1994, pp. 499-532.

Edwards, W.K. and Mynatt, E.D. An Architecture for Transforming Graphical Interfaces. In the *Proceedings of UIST'94: User Interface Software and Technology Symposium*, Marina Del Ray, CA., Nov. 2-4, 1994, 39-47.

Gaver, W. W. Everyday listening and auditory icons. Doctoral Dissertation, University of California, San Diego. 1988.

Gaver, W.W. Using and Creating Auditory Icons. In *Auditory Display: Sonification, Audification and Auditory Interfaces*, edited by G. Kramer, SFI Studies in the Sciences of Complexity Proc. Vol. XVIII, Addison-Wesley, 1994, pp. 417-446.

Mynatt, E. and Weber, G. Nonvisual Presentation of Graphical User Interfaces: Contrasting Two Approaches," in the *Proceedings of the ACM Conference on Human Factors in Computing Systems*, 1994.

Mynatt, E. Auditory Presentation of Graphical User Interfaces, In *Auditory Display: Sonification, Audification and Auditory Interfaces*, edited by G. Kramer, SFI Studies in the Sciences of Complexity Proc. Vol. XVIII, Addison-Wesley, 1994, pp. 533-556.

Mynatt, E. Evaluating Mercator. Submitted to CHI '97.

Sikora, C., Roberts, L. and Murray L. T. Musical vs. Real World Feedback Signals. In *CHI '95 Conference Companion, Human Factors in Computing Systems*. ACM. 1995, pp. 220-221.