CENTER FOR COMPUTER RESEARCH IN MUSIC AND ACOUSTICS
OCTOBER 1995

DEPARTMENT OF MUSIC
REPORT NO. STAN-M-95

THE MOOSE: A HAPTIC USER INTERFACE FOR BLIND PERS WITH APPLICATION TO THE DIGITAL SOUND STUDIO

RICHARD BRENT GILLESPIE
SILE O'MODHRAIN

RESEARCH SPONSORED BY
THE CCRMA INDUSTRIAL AFFILIATE PROGRAM,
THE STANFORD OFFICE OF TECHNOLOGY LICENSING
AND THE FULLBRIGHT SCHOLARSHIP PROGRAM

CCRMA
DEPARTMENT OF MUSIC
STANFORD UNIVERSITY
STANFORD, CALIFORNIA 94305-8180

The Moose: A Haptic User Interface for Blind Persons

Sile O'Modhrain and Brent Gillespie
Center for Computer Research in Music and Acoustics (CCRMA)
Stanford University
sile@ccrma.stanford.edu, brent@ccrma.stanford.edu
October 16, 1995

Abstract

This paper presents our work to date on a haptic interface whose immediate aim is to provide access for blind sound engineers to the graphics-based computer interfaces currently found in digital sound studios. We describe the hardware and supporting software which together reinterprets a Microsoft Windows screen for the haptic senses.

With a haptic interface, screen objects such as buttons, sliders and pull-down menus are presented mechanically to the user's haptic senses (kinesthetic and tactile), where they can be felt, located, identified, and, through the use of the same device for input, activated. We have built a prototype two-axis device which operates much like a mouse, except that it is also able to move under its own power. Using this prototype device we have already implemented an interface for Microsoft Windows and have proven the feasibility and usefulness of the haptic interface approach for non-visual computer access. We expect that haptic interface devices will become standard computer interface tools, supplementing the visual presentation with haptic presentation for all users. A more holistic presentation of information will be made by the computer. This approach is, we believe, particularly valuable in the design of application interfaces for digital audio editing. We have implemented a prototype sound file editing application in which the sound waveform is presented haptically as a contoured groove within the workspace.

1 Introduction

'Haptics' refers to the human tactile (cutaneous) and kinesthetic (muscle movement) senses. A haptic interface is a computer-controlled motorized device to be held in the hand by a user, which displays information to that user's haptic senses. It is an extremely powerful modality for interface design because the same device can be used for both displaying output from the computer and accepting input from the user. Moreover, using haptics significantly reduces the burden on other information channels such as vision and audition, thereby freeing these channels for other tasks. In certain instances it is, we believe, possible to completely substitute haptics for other sensory modalities. In this way, graphical information displayed on a computer's screen can be made accessible to blind persons who at the moment are denied access to standard Graphical User Interfaces (GUIs). For example, by producing forces on the user's hand which are a function of both the user's motion and the screen contents, virtual objects can be created.

In particular, we are interested applying haptics as a sensory substitute for the graphical interface of today's digital audio editing applications. A sound engineer's audio channel is already occupied while using such applications, so our display cannot make good use sound or speech output. Our eventual goal, therefore, is to design a display which only uses haptics. The most natural format to adopt, especially when we wish to grant access to existing third-party applications, is the 'powered mouse', since it so easily adapts to the mouse paradigm. Our mouse, however, also functions as an output device, taking over the role of the screen. Our use of haptics for audio editing effectively brings us back to the analogue audio studio, where sound editing was primarily a hands-on affair–a move which will be welcomed by many a modern sound engineer.

A handful of other research groups are working on powered-mouse type interfaces for GUIs. In Wisconsin, the TRACE Group is developing a mouse with both vibrotactile and force feedback has been developed. [Akamatsu, Sato 1994] At Stanford in the 1970s the Optacon was developed by John Linvill, and stands as one of the first commercialized haptic display devices. [Linvill 1973] Other access devices for the blind and deaf-blind based on haptics have also been developed [Loubal 1992], [York 1989] [Frisken-Gibson 1987] [Eberhardt 1993] [Kelly, Salcudean 1994] See [Minsky 1995] for a comprehensive overview of current haptics technology.

Our project is somewhat unique in that it must pay very careful attention to the manner in which information is displayed through the audio channel. Our constant directive is to substitute visual presentation with haptic presentation alone.

In section 2, we introduce the architecture of our prototype interface. We document the design of a Haptic User Interface (HUI) which translates certain elements of the standard Graphical User Interface (GUI) into a form where they can be felt. Section 3 details the specific goals for which this research was undertaken. Section 4 summarizes.

2 Our Prototype Interface

We have designed and built a working prototype which serves to prove the concept and has generated quite a bit of enthusiasm from those who have seen (felt) it. It is basically a powered mouse, giving the user the ability to feel the screen objects under the mouse cursor. This mouse is used to navigate the screen like a regular mouse; but by reflecting forces (produced by its motors) back to the user, it presents a haptic representation of the symbols on the screen. In other words, as the powered mouse is moved, its position is continuously compared against the screen image. If the mouse should alight on an icon, a haptic representation of that icon will be presented by the motorized mouse for the user to feel and explore. The edge of a window, for example, might feel like a detent under the mouse. A button might feel like a patch of sandpaper. Once a desired icon has been found, it may be selected using a typical mouse button. A user can explore a screen, activate 'buttons' from menus, and select other screens which in turn will be mapped and haptically displayed. For text which appears on the screen (file names, button labels, etc.) we have experimented with integrating a speech synthesizer into the interface. We are exploring the presentation of altogether new information through haptics. For example, the next appropriate user action in a given context can be indicated by causing the powered mouse to gravitate to a particular icon.

2.1 Hardware

Figure 1 shows the hardware components of our present planar haptic interface which we affectionately call 'The Moose'. The puck or manipulandum in the center is coupled to two linear voice coil motors through two perpendicularly oriented flexures.

The unique feature of our hardware design is this double flexure. On the present prototype, the double flexure is executed in two pairs of foot-long strips of spring steel. The double flexure conveniently decouples the 2-axis motion of the puck into two single-axis motions at the linear motors. Moments and vertical forces are resisted, yet translations in the horizontal plane are transmitted directly to the motors by the manipulandum. The kinematics of this device are simple and very nearly linear, making forward and inverse kinematic calculations unnecessary. Furthermore, the workspace is flat, square like a mousepad, and free of singularities. The entire workspace is also naturally counterbalanced. This design ensures that very little inertia is added to the motors. Over-limit forces will cause buckling of the flexures, which we consider a safety feature. The only real disadvantage of the double flexure design is added high-frequency structural resonances inherent in the flexures themselves. These resonances will bandlimit the display capabilities. But if chosen high enough by design, they should not interfere with the bulk of haptic object images.

A simple Digital I/O card provides for PC-bus communication to four 12-bit DACs and four quadrature counters. The voltage outputs of two DACs, ranging +/- 5 Volts, feed to two transconductance amplifiers based on the LM12 power op amp and in turn to the motors. A linear position encoder, 150 lines per inch, reads position on each of the motors while the count circuit maintains an up-to-date binary representation of position. Other digital switch inputs such as buttons can be polled from software. Finally, a speech synthesizer linked through the serial port is available.

Future hardware enhancements will include the following: 1) a braille display to take the place of the speech synthesizer for text output, 2) the use of braille cells for "shape" display, and 3) the use of a small voice coil motor for vibration and texture display [Kontarinis 94].

2.2 Software

Various control routines which create haptic effects such as virtual springs, textures, and buttons have been developed and incorporated into our Windows interface. By combining these primi-

Windows icon. For example, our haptic checkbox has a frame surrounding the checkbox text and a detent corresponding to the checkbox state indicator. Just as the state indicator changes color when the checkbox is checked, so also our haptic checkbox state indicator changes from a detented spring to a repelling pyramid which is immediately apparent when the profile of the checkbox is examined.

Figure 2 is an outline of the architecture of our software. Our software is divided into three distinct modules:

2.2.1 The Icon Management Class

This module is responsible for "mapping" each new screen as it appears and storing information, such as icon dimensions and icon names, about each icons it finds in a linked list.

2.2.2 The Hapticon Management Class

This module queries the icon manager, using the obtained information to construct a list of corresponding hapticons. It encapsulates the haptic properties of each hapticon in control laws and lookup tables for convenient use by the control module.

2.2.3 The Control Module

The control module is responsible for executing the control loop. Its action is embedded in the Windows message loop. It polls the current moose position, moose button status, and current window identifier. If the current window has changed, it initiates the mapping of the new window by the icon manager and requests the hapticon manager to update its hapticon list. The control module constantly passes the current moose position to the hapticon manager and receives a force appropriate for whatever icon lies at that position. The control module then outputs that force to the moose. Moose button clicks and moves are passed through to the Windows mouse button control routine.

3 Specific Goals

We are currently experimenting and developing a palette of haptic effects which will be used to explore and allow comparisons among various haptic substitutes for graphic objects- i.e. detents for buttons, solid blocks for inaccessible objects, compliant and non-compliant borders for windows, and so on. We hope that this research will result in a characterization of graphic interface objects

time we hope that a common practice for haptic interface design will arise.

The project has attracted the interest of Neil Scott and his team at Stanford's Center for the Study of Language and Information, who are eager to incorporate our work into their Total Access Port (TAP) system. TAP is aimed at developing a generic adaptive interface port through which interface device signals can be intercepted and hence made available to whatever access device a disabled user finds most appropriate. The system's broad goal is to provide the individual with one personalized interface which they can bring with them to whatever computer they need to use. The haptic interface is a very realistic option for conveying the contents of a GUI to a blind person. Scott's team will substitute our icon management module with their own vision recognition routines which obtain the icon information directly from the video signal rather than from the Windows environment. The advantage of this system is that it will be platform-independent and will thus allow blind people to use the same access device for any number of computers running any type of operating system.

Having proven the feasibility of substituting haptics for graphics in a prototype of a general computer user interface for the blind, we now turn our attention to designing a haptic interface for sound processing applications.

Our project's goal for some time has been to provide the ability to cursor through an audio stream as one does with a tape head on magnetic tape. We are now in fact able to both haptically explore and hear the sound signal simultaneously. The contour of the recorded sound on the tape is felt as a virtual groove embedded in the workspace while the same signal under the hand is heard. We have most recently added a third axis to our display platform for the purpose of playing back the sound signals being edited. By using the DACs already available in our haptic display hardware, we sidestepped the need to develop the requisite realtime audio tools for the PC. Existing PC sound cards are simply not capable of real-time audio processing. Also, we require more than just sound triggering, which is all that MIDI driven synthesizers could have provided.

We now have a system which duplicates much of the analogue tape machine interface. It allows for the display of a soundfile both haptically and graphically and allows one to cursor through the sound, examining it's contour in any one of four "zoom" levels. The user may 'throw' the virtual tape, which will continue to move past their hand as if it had inertia. See [Gillespie 93] for a discusplayed back. The user may drop marks and feel for previously marked passages, and perform cut cut and splice operations in an intuitive manner. But beyond this, we have programmability –the opportunity to explore transformations between the haptic and audio presentations of the media under scrutiny. In many respects, we can do better than the analogue tape machine. By implementing a looping-buffer process, we can cursor through the signal while allowing for dynamic relationships between the tape velocity and sounded pitch. When cursoring stops, for example, the output loops on the segment of audio lying under the cursor at its real pitch. Moreover, the speed at which the cursor passes over the signal in this case only determines the streaming speed and not the pitch of the audio output.

Other types of pre-processing of the audio signal would, we believe, be worth exploring. An audio event detector [Chafe 1986] could be used to place haptic "landmarks" on the events of special interest in the signal. For example, speech, silences, tone onsets, and decay anomalies can all be given special haptic characteristics. Other confusing content could be screened out.

4 Summary

Haptic technology is particularly well suited to solving the problem posed by this project –namely to make digital sound editing applications accessible to blind sound engineers. Unlike existing access technologies for the blind, most of which rely heavily on speech output, it makes no demands on the auditory channel. The ears are left free for the task of editing sound.

A new motorized mouse takes over the function of the conventional mouse while conjoining the output role the screen. With our powered mouse, a blind user can navigate and interact through application's window. Sighted users may also realize advantages in speed and dexterity.

Using Windows Internals, we have transcribed the visual information of the screen and made it available to the haptic senses. The Windows environment enables inquisitive software such as ours to access all information on the screen. Our software simply gathers that information and displays it haptically. The real advantage of our haptic interface over a speech screen reader is that information about the icon and window topology is presented directly and immediately rather than through time-consuming descriptive language.

We have explored the new opportunities afforded by haptic display within a prototype audio editing environment. We have used the old principles could be used to effectively display the information in graphs and diagrammatic figures. We foresee that in such applications, haptic interface technology will not only help blind people but will also supplement graphics for sighted computer users.

References

[Akamatsu, Sato 1994] Akamatsu, M., Sato, S. and MacKenzie, I.S. *Multimodal mouse: A mouse-type device with tactile and force display* Presence, Vol 3 No. 1, Winter 1994 pp 73-80.

[Chafe 1986] Chafe, C. and Jaffe, D. *Source separation and note identification in polyphonic music.* IEEE Conference on Acousics, Speech, and Signal Proccesing, Vol 2, 1986, pp. 1289-1292.

[Eberhardt 1993] Eberhardt, S.P., et al. *OMAR: a haptic display for speech perception by deaf and deaf-blind individuals.* Proceedings of IEEE Virtual Reality Annual International Symposium. Held: Seattle, WA, USA, 18-22 Sept. 1993. p. 195-201

[Edwards 1989] Edwards, A.D.N. *Soundtrack: an auditory interface for blind users.* Human Computer Interactions vol.4, no.1, 1989 p. 45-66.

[Frisken-Gibson 1987] Frisken-Gibson, S.F., et al. *A 64-solenoid, four-level fingertip search display for the blind.* IEEE Trasactions on biomedical engineering. (Dec. 1987) vol.BME-34, no.12, p. 963-5.

[Gillespie 93] B. Gillespie, M. Cutkosky. *Interactive dynamics with haptic display* In Proceedings of the 1993 ASME Winter Annual Meeting, New Orleans, pp. 65-72.

[Kelly, Salcudean 1994] Kelly, A.J., and Salcudean, S.E. *On the development of a force-feedback mouse and its integration into a graphical user interface.* in Proceedings ASME Winter Annual Meeting, DSC-Vol. 55-1, 1994.

[Kontarinis 1994] Kontarinis, D. A. and Howe, R. D. *Tactile display of vibratory information in teleoperation and virtual environments* Presence, in press.

[Linvill 1973] Linvill, J. *Research and development of tactile facsimile reading aid for the blind: the optacon* [Washington] U.S. Dept. of Health, Education and Welfare, Office of Ed

[Loubal 1992] Loubal, P.S. *Fingertip maptracing devices for the blind.* Proceedings of Conference on Technology and Persons with Disabilities. Held: Los Angeles, CA, USA, 18-21 March 1992. p. 315-18

[Minsky 1995] Minsky, M. *Computational Haptics: Texture* PhD Thesis, MIT Media Labs, 1995.

[Wiker 1991] Wiker, S.F., et al. *Development of tactile mice for blind access to computers: importance of stimulation locus, object size, and vibrotactile display resolution.* Proceedings of the human factors society 35th annual meeting. vol.1 Held: San Francisco, CA, USA, 2-6 Sept. 1991. p. 708-12

[York 1989] York, B.W., et al. *Tools to support blind programmers.* 17th Annual ACM Computer Science Conference. Held: Louisville, KY, USA, 21-23 Feb. 1989. p. 5-11

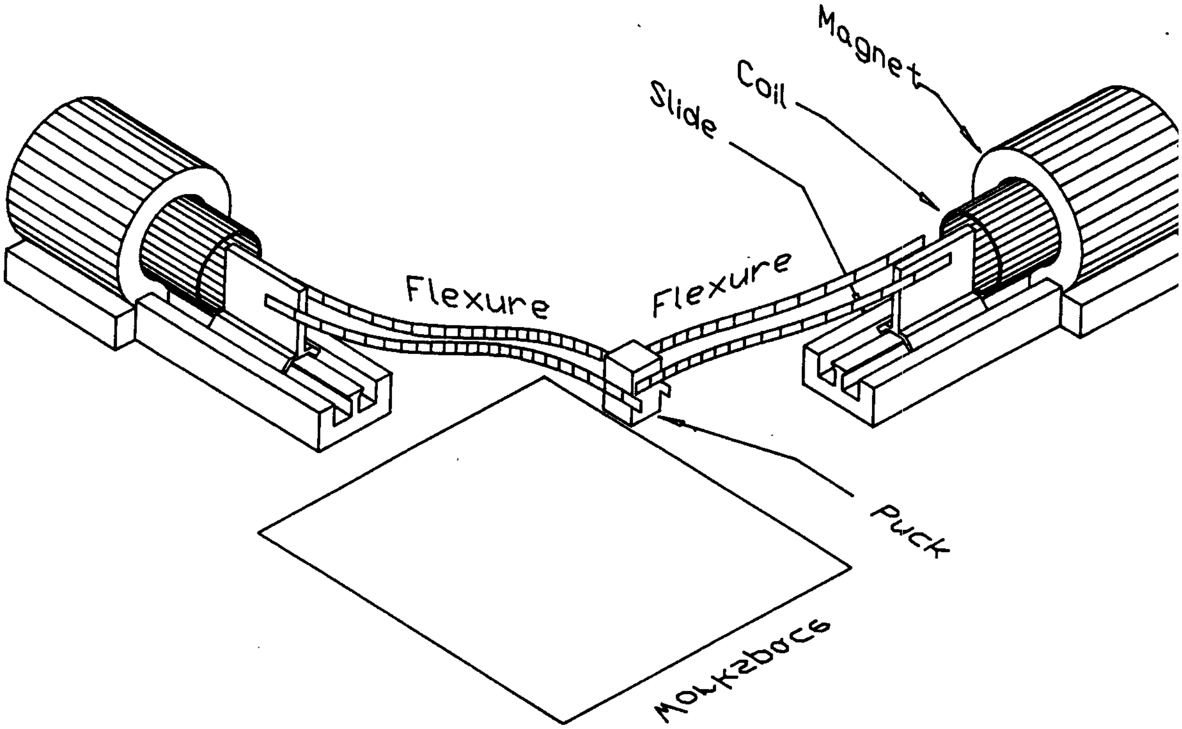
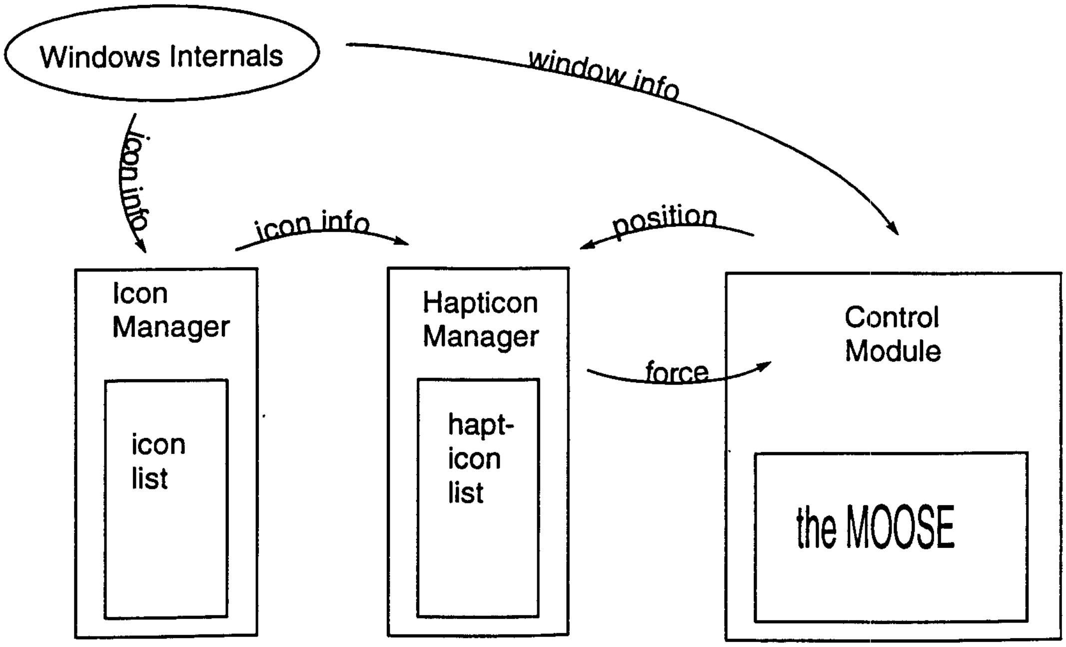


Figure 1. The Moose



A Haptic Interface for the Digital Sound Studio

Sile O'Modhrain and Brent Gillespie
Center for Computer Research in Music and Acoustics (CCRMA)
Stanford University
sile@ccrma.stanford.edu, brent@ccrma.stanford.edu
May 4, 1995

Abstract

This project involves the development of a new computer interface based on haptics. Information will be transferred from computer to user mechanically (by touch) instead of visually or aurally. The immediate aim is to provide access for blind sound engineers to the graphics-based computer interfaces currently found in digital sound studios.

Access for blind persons to computer applications with graphical user interfaces is virtually nonexistent at present. A mouse is of no use without the user's visual tracking of the cursor on the screen. With a haptic interface, screen objects such as buttons, sliders and pull-down menus will be presented mechanically to the user's haptic senses (kinesthetic and tactile), where they can be felt, located, identified, and, through the use of the same device for input, activated. We have built a prototype two-axis device which operates much like a mouse, except that it is also able to move under its own power. By producing forces on the user's hand which are a function of both the user's motions and the buttons or windows under the cursor, touchable representations of the screen objects are created. Using this prototype device we have already proven the feasibility and usefulness of the haptic interface approach for non-visual computer access. We expect that haptic interface devices will become standard computer interface tools, supplementing the visual presentation with haptic presentation for all users. Less attention to visual tracking would be needed. A more complete and holistic presentation of information will be made by the computer. This approach is, we believe, particularly valuable in the design of application interfaces for digital audio editing.

1 Outline

This paper is divided into the following five sections. First we introduce the project and its goals. In Section 2, we document the design and development of a Haptic User Interface (HUI) which translates certain elements of the standard Graphical User Interface (GUI) into a form where they can be felt. We describe how these haptic icons are 'displayed' using a prototype powered mouse. Section 3 details the very specific goals for which this research was undertaken - namely to make digital sound editing applications (which are heavily dependent on the GUI) accessible to blind sound engineers. Section 4 discusses aspects of a working perceptual model for sound editing and its relationship to the tactile and kinesthetic aspects of manipulating existing sound media. Finally in section 5, we explore ways in which haptic tech through touch. Steps toward incorporating these tools into the digital sound studio will also be discussed.

2 Introduction

'Haptics' refers to the human tactile (cutaneous) and kinesthetic (muscle movement) senses. A haptic interface is a computer-controlled motorized device to be held in the hand by a user, which displays information to that user's haptic senses. It is an extremely powerful modality for interface design because the same device can be used for both displaying output from the computer and accepting input from the user. Moreover, using haptics significantly reduces the burden on other information channels such as vision and audition, thereby freeing these channels for other tasks. In ities. In this way, graphical information displayed on a computer's screen can be made accessible to blind persons who at the moment are denied access to most standard Graphical User Interfaces (GUIs).

Haptic technology is particularly well suited to solving the problem posed by this project - namely to make digital sound editing applications accessible to blind sound engineers. Unlike existing access technologies for the blind, most of which rely heavily on speech output systems, it makes no demands on the auditory channel. The ears are left free for the task of editing sound.

We are aware of several other research groups working on powered-mouse type interfaces for GUIs. At the Technical University Berlin, the "GUIB" group is addressing access to Computer Aided Design tools with a motorized trackball. [TU-Berlin] In Wisconsin, the TRACE Group is developing a computer access tool for the blind based on haptics. [Wiker 1991] At the University of Toronto, a pantograph device is being applied as a substitute for the mouse. [Toronto] At the university of Tokyo, a mouse with vibrotactile and force feedback has been developed. [Akamatsu, Sato 1994] At Stanford in the '70s the Optacon was developed by John Linvill, and stands the one of the first commercialized haptic display devices. [Linvill 1973] Other access devices for the blind and deaf-blind based on haptics have also been developed [Loubal 1992] , [York 1989] [Frisken-Gibson 1987] [Eberhardt 1993] [Kelly, Salcudean 1994]

See [Minsky 1995] for a comprehensive overview of current technology in haptics.

3 Our Prototype Interface

We have designed and built a working prototype which serves to prove the concept and has generated quite a bit of enthusiasm from those who have seen (felt) it. It is basically a powered mouse, giving the user the ability to feel the screen objects under the mouse cursor. This mouse is used to navigate the screen like a regular mouse; but by reflecting forces produced by its motors back to the user, it gives a haptic representation of the symbols on the screen. In other words, as the powered mouse is moved, its position is continuously compared against the screen image. If the mouse should alight on an icon, a haptic representation of that icon will be presented by the motorized mouse for the user to feel and explore. The edge of a window, for example, might feel like a detent under the mouse. A button might feel like a patch tons' from menus and select other screens which in turn will be mapped and haptically displayed. For text which appears on the screen (file names, button labels, etc.) we have experimented with integrating a speech synthesizer into the interface. Future explorations will include a braille display.

3.1 Hardware

Figure 1 shows the hardware components of our present planar haptic interface which we affectionately call The Moose. The puck or manipulandum in the center is coupled to two linear voice coil motors through two perpendicularly oriented flexures.

The unique feature of our hardware design is this double flexure. On the present prototype, the double flexure is executed in two pairs of foot-long strips of spring steel. The double flexure conveniently decouples the 2-axis motion of the puck into two single-axis motions at the linear motors. Moments and vertical forces are resisted, yet translations in the horizontal plane are transmitted directly to the motors. The kinematics of this device are simple and very nearly linear, making forward and inverse kinematic calculations unnecessary. Furthermore, the workspace is flat and square like a mousepad, and free of singularities. The entire workspace is naturally counterbalanced. This design ensures that very little inertia is added to the motors. Over-limit forces will cause buckling of the flexures, which we consider a safety feature. The only real disadvantage of the double flexure design is added high-frequency structural resonances inherent in the flexures themselves. These resonances will bandlimit the display capabilities. But if chosen high enough by design (greater than about 20Hz,) they should not interfere with the bulk of haptic object images.

There are two extensions to the double flexure idea which we plan to explore. First, an additional flexure along with the substitution of rods instead of strips for all flexures will allow for 3-axis motion. For such a design, counterbalancing would be necessary on the added vertical axis. Second, the use of three independently driven strips, instead of four driven in pairs, along with pivots at the puck will provide for just-constrained drive in all three planar degrees of freedom, the two displacements and one rotation.

Double flexures have been used in electron lithographic machines and other small-motion robotic devices. But as far as we are aware, ours is the first incorporation of flexures into a haptic interface. Planned improvements to our mechansmall rotary motors. Higher resolution position encoding will also be a high priority.

Figure 2 is a block diagram showing major components of the drive system. A simple 196-bit Digital I/O card provides for PC-bus communication to two simple home-grown circuits, one a 12-bit DAC and the other a quadrature counter. The voltage output of the DAC, ranging +/- 5 Volts, feeds to a transconductance amplifier based on the LM12 power op amp. This linear amplifier is configured with a gain of 1, so that 1 volt drives 1 amp through the motor. A linear position encoder, 150 lines per inch, reads position on each of the motors and the 16-bit quadrature decode and count circuit maintains an up-to-date binary representation of position always ready for sampling by the software. Other digital switch inputs such as Moose buttons can also be polled from software. Finally, a speech synthesizer linked through the serial port is available as well as a MIDI synthesizer.

3.2 Software

Various control routines which create haptic effects such as virtual springs, textures, and buttons have been developed and incorporated into a simple non-Windows GUI on the PC. The control program is divided into three very distinct modules:

I: A set of menus and pages which provide graphical access to some simple operations - controlling a MIDI Synthesizer, querying DOS time, date and disk-space functions etc.

II: The Vision Recognition module (ScreenMapper), independent of the GUI, scans video memory and determines the shape, color and location of objects on the screen. The ScreenMapper then stores these parameters, and retrieves them when queried by the haptic display handling module.

III: The Haptic Display (Moose) module controls the hardware, continuously tracking the Moose cursor's current position and comparing it to the stored screen representation.

Figure 3 presents a typical page of our present graphical user interface featuring haptic access. The software used to create this screen and others like it was built upon a source-code library from the book *Graphical User Interfaces with Turbo C++*, by Ted Faison [Faison 1991]. From this library, interface elements such as buttons, sliders, and pop-up windows are available.

Figure 4 is a depiction of the force gradients which are overlaid by the Moose on its workspace for this page, giving rise to the virtual buttons.

center when it encounters the area corresponding to a button.

Figure 5 is an Inheritance diagram of the various classes in use at present. As stated above, the module called the ScreenMapper reads the screen after it has been drawn and stores away the information where it can be quickly accessed by the Moose module. The Moose is otherwise coupled into the software like the module which manages the Mouse. The NewEvents module and the HapticDisplay module are global objects, accessible from any other part of the software. They can message freely back and forth between each other.

We are exploring virtual object creation both in the form of control laws and lookup tables and hybrids of the two. The message loop (and by extension the control loop) starts with a polling of the Moose position and an update of the cursor position. That position is used to check against the map of the screen for the existence of a button under the cursor. If a button is found, a force representation (from among several options) is calculated and fed back out to the driving motors of the Moose. In this manner, the Moose sends out forces in response to incoming motion, and does so dynamically, thus creating programmable myriad mechanical impedances.

4 Specific Goals

We are currently experimenting and developing a palette of haptic effects which will be used to explore and allow comparisons among various haptic substitutes for graphic objects- i.e. detents for buttons, solid blocks for inaccessible objects, compliant and non-compliant borders for windows, etc. We hope that this research will result in a characterization of graphic interface objects and that a corresponding library of haptic effects for representing these objects will be created so that in time, a common practice for haptic interface design will arise.

Results so far are the product of about seven month's work. Involved are Sile O'Modhrain and Brent Gillespie, under the direction of Professor Chris Chafe (Music) and with the aid of Professor Mark Cutkosky (Mechanical Engineering). The work is funded this year by a Stanford Office of Technology Licensing Research Incentive Grant. Sile O'Modhrain was initially at Stanford as a Fulbright Scholar from Ireland and is now a graduate student in Computer Music Theory. Brent Gillespie, a graduate student in Mechanical Engineering specializing in haptic interface design began collaborating with Sile on the development of this cations. She had worked for the BBC as an audio engineer until the introduction of GUI-outfitted digital studio tools made it impossible for her, as a blind sound engineer, to function as she had in their analog studios. She therefore decided to combine past experience in both software design and sound engineering and came to Stanford with a view to addressing this problem.

The project has also attracted the interest of Neil Scott and his team at CSLI who are eager to incorporate our work into their Total Access Port (TAP) system. TAP is aimed at developing a generic adaptive interface port where interface device signals can be intercepted and hence made available to whatever access device a disabled user finds most appropriate. The system's broad goal is to provide the individual with one adapted interface which they can bring with them to whatever computer they need to use. The haptic interface is, they believe, a very realistic option for conveying the contents of a GUI to a blind person.

Our project's grail for some time has been the ability to cursor through an audio stream as one does with a tape head on magnetic tape. We would like be able to both haptically explore and hear the signal simultaneously. The contour of the recorded sound on the tape will be felt while the signal under the hand is heard. To this end, we have most recently added a third axes to our display platform, for the purpose of playing back the sound signals being edited. By using the same DACs we had built for driving our haptic display, we sidestepped the need to stop and develop the real-time capable audio tools we need. Existing sound cards for the PC are simply not capable of real-time audio processing. Also, we require more than just sound triggering, as is available from MIDI driven synthesizers, in order to implement virtual magnetic tape.

We now have a system which duplicates much of the analogue tape machine interface. But beyond this, we have programmability, the opportunity to explore transformations between the haptic and audio presentations of the media under scrutiny. In many respects, we can do better than the analogue tape machine. By implementing a looping-buffer process, we can cursor through the signal while allowing for dynamic relationships between the tape velocity and sounded pitch. When movement stops, for example, the output would loop on the segment of audio lying under the cursor at its real pitch. Moreover, the speed at which the cursor passes over the signal would in this case only determine the streaming speed and not the pitch of the audio output.

Other types of pre-processing of the audio sig-

the events in the signal of special interest. For example, speech, silences, tone onsets and decay anomalies can all be given special haptic icons. Other confusing content could be screened out [Chafe 1986]. wWe are be exploring mappings from information to two sensory modalities, haptic and audio. Furthermore, we are exploring the optimality of those mapping with respect to the editing needs of the sound engineer.

Having proven the feasibility of substituting haptics for graphics in a prototype of a general computer user interface for the blind, we now want to turn our attention to designing a haptic interface for sound processing applications.

5 Haptics and Sound

Here we suggest that working perceptual models for contemporary digital sound editing tools are closely related to and usually based on the haptic aspects of manipulating the sound media itself. We propose that haptic interface is an opportunity to take these essentially mechanical metaphors and more fully exploit them.

The fact that creating sound is a fundamentally mechanical phenomenon leads us to believe that a haptic interface to sound editing applications is even more promising than a visual interface. Musical instruments are generally more interesting to feel than they are to look at. Most importantly, it is through mechanical interaction with an instrument that we learn to make music. It stands to reason, therefore, that mechanical presentation of audio information will inspire natural and intuitive means for its manipulation.

One rarely thinks of the process of sound editing as a musical performance. Likewise, the process of composition is uniquely distinct from performing. However, activities such as improvisation do indeed bridge the gap between composition and performance. We believe that there exists a similar gray area between the roles adopted and the tools used by the sound engineer and the performing musician. This gray area is ready for exploration. The relationship between the audio engineer and the instruments they use should be allowed to expand and evolve, even encroach upon the territory of the live performance artist.

In recent years, the increasing commercial focus on developing alternative interface devices to supplement - or in some cases entirely replace - the ubiquitous keyboard and mouse amply testify to an increasing need for more task-specific interface tools. CAD systems, for example, are now being provided with pen pads to allow the user to believe that the sound engineer should be provided with tools that bear a more realistic relationship to the kinds of tasks they must perform.

Sound processing includes two basic elements: mixing and editing. Mixing has to do with combining sounds and editing has to do with separating them. Thus the term mixing covers a wide range of operations: equalization, effects processing, cross-fading, and other signal processing operations. Figure 6 presents a taxonomy of sound engineering tools. This is just one possible division, but nonetheless a starting point for analysis. Other divisions could be made along the time/frequency domains. We shall address the haptic aspects of mixing and editing with specific examples in the following two sub-sections.

5.1 Mixing

If a sound source within a mix is too loud, the engineer must be able to effortlessly reach out and turn it down. On the board of a well-thought out mixing session, the engineer will ensure that all sources are grouped and channelled in such a way that their controls are always within easy reach. Thus the task of turning down one element within a mix has been reduced to a single hand movement: reach out and grab the slider that controls the sound's level and pull it towards you. In this way, learning to control a mixing board becomes a process which is intricately based on muscle memory much like learning to play an instrument. The problem with the generic mouse/keyboard interface for mixing applications is that this kinesthetic aspect of sound control is missing. If the level of one sound within a mix is to be reduced, the engineer must move the mouse to point at the appropriate slider's icon, select it and drag it down along its graphic track. To break this problem down further:

1: The mouse will have to be picked up before anything can be done.

2: The mouse's location and hence the kinesthetic point of departure will be random, depending on where the mouse was left after the last task. (The mouse is a relative position pointing device, not absolute.)

3: The engineer must employ visual tracking to locate the mouse correctly on what is often a very crowded screen.

4: The size of the physical motion required to decrease a signal's level is no longer constant (as on a physical mixing board) but depends upon the size of the computer's monitor and the window's zoom setting which may be recurringly changing.

5: The tactile and kinesthetic cues (the feel of age of these controls. Their haptic aspects are no longer an aid to memory because the mouse feels the same whatever the task - turning, sliding, etc.

5.2 Editing

But if the absence of haptic cues in digital mixing applications makes the job of mixing harder, the lack of haptic feedback is an even greater loss when it comes to directly editing sound. Even the vocabulary of editing –cutting, splicing, out-takes, off-cuts, reel-rocking, etc. - is itself directly related to the physical actions required to perform these various tasks. While it is unreasonable to expect computer-based editing applications to mimic the systems they were intended to improve upon, it is also counterproductive to place upon the sound engineer the burden of translating the highly physical metaphor of the editing application into the kinds of gestures associated with the keyboard and mouse of the GUI. In other words, the metaphor says "cut here" and "splice there", yet the tasks are actually performed by moving a plastic mouse around. The mouse and interface thereby creates a cognitive dissonance.

For all their draw-backs, the standard reel-to-reel tape machines which were the back-bone of analogue recording studios had one very strong plus. Their mechanical design placed the sound engineer in a very physical relationship with the sound being edited. Edit points were fine-tuned by placing one hand on each reel of tape and rocking the tape against the machine's play-back head. The size and speed of the hand and arm movements directly controlled the character of the sound heard. Movements were varied depending on the type of sound being edited - sharp attacks required short, jerky movements to accurately define their start-points while sounds with gradual attacks required a much slower arm movement to separate their beginnings from background noise. Once an out-point had been marked and cut, the engineer usually pulled the tape past the playback heads listening for the desired "in point". This was a very useful technique because the speed at which the tape was pulled directly controlled the pitch and hence the intelligibility of the sound. The skill rested in being able to listen at very high speeds for clues and at the same time coordinate hand movements. Basically, development of such skill depends on availability of fine motion control, which translates into close physical communication between the recording medium and the sound engineer.

This ability to audibly scan and very directly control scan speed is perhaps the most noticethis facility by designing peripheral control panels which reintroduce standard tape-machine controls (play, stop, rewind, etc) and shuttle wheels. However very few of these devices employ meaningful haptic cues. One digital unit which has a shuttle wheel is the high-end Panasonic DAT machine. The wheel has a rest position which is in the center of a sprung detent. When a tape is playing, the operator can scan backwards and forwards through the sound by turning this shuttle wheel in the appropriate direction. The further from the center position the wheel is pushed, the greater the spring's resistance and the faster the tape is scanned. Though this is a very simple device, it is a rather atypical example of an attempt to re-introduce haptics into digital sound control. We have yet to find even this simple level of haptic cuing in any currently available computer-based sound editing system. Many, it is true, now incorporate some form of jog/shuttle wheel on a peripheral device, but these wheels do not convey any meaningful haptic information. In most systems, they are extremely low-friction devices which can be very easily knocked off position and which require multiple rotations to cycle through any reasonably-sized section of sound material. The sound feed-back, too, is often chunked so coarsely that it is difficult to fine-tune an edit point. To finally home in on an edit point, therefore, the engineer has to suddenly switch from listening to a sound to viewing its spectrum or waveform on the screen and selecting the point entirely on the basis of visual, not audio criteria.

The mechanical aspects of digital tools require special consideration from their designers. Consumers have historically demanded such attention from their analogue tools and would demand such from their digital editing tools if they thought they could. Small points of mechanical design finesse can be identified as reason for domination of the market by certain products. Perceived quality in of a tool depends to a very large degree on the feel, not just the look. Audio engineers, like other musicians, are intensely aware of the physical feel of a device and the implemented mapping of input to effect.

5.3 Summary: Haptics and Sound

With a few simple haptic interface tools we believe that we can bring back to the editing process some of its former intuitiveness and flexibility. We can design haptic controls, virtual or real, to correspond to graphic icons. We can control their "feel" by making them more or less resistant to being moved. A shuttle wheel detent is

able in the real world of the analogue audio studio. The linear, log, or other scaling of a slider, for example, might be made apparent and even readily configurable to a user by programmably endowing the slider with suggestive resistance profiles. The record-enable buttons on a virtual multi-track machine could be constructed in such a way that, when they were set to "disabled" they would be made to feel like solid blocks, repelling the hand and therefore preventing the operator from even being able to point at them unintentionally and record over a previously recorded track.

The mechanical aspects of analogue tools are diverse. Noteworthy is that the audio engineer enjoys close contact with the recording media. The unique physical operations on the media persist today as metaphors in digital audio editing tools. Also, each recording medium affords a certain kind of sound manipulation. For example, records are scratched back and forth, tapes are slowed and sped up for wow, flutter, and other time/pitch effects. Indeed, the musical arts themselves are strongly influenced by the recording technologies (and the recording interface technologies) of the day. Brassage, Musique Concrete, RAP and House Music forms and other musical genres are broad examples of technologically influenced music. We look forward to enjoying the artistic output inspired by the more programmable yet multimodal and physically coupled interfaces of the future which will feature haptic components.

6 Beyond Machines

By far our greatest hope in this project is, however, to explore ways in which we can entirely divorce the process of editing sound from the paraphernalia of the audio studio. This is not just a hope but, with the advent of Virtual Reality and hence the need to process 3-dimensional sound in conjunction with 3-dimensional graphics, is something which has become a real need. If sound is to exist in a 3-dimensional space then it is necessary to provide an environment where the sound engineer can, like a sculptor, build up and mould elements within the sound space. Haptics provides us with a ideal interface for this process because it reduces the need to visually monitor the sound processing application, leaving the engineer free to concentrate on the sound itself. Moreover, since haptic objects are virtual objects, they are infinitely variable and can relate in a very flexible way to elements of the sound material being edited. They can be suspended in free-space and be pushed around and regrouped like building with a press of the thumb. The energy expended to squash the spikes will make the process downright satisfying. The need to develop an abstract conceptual model for mapping the complex variables associated with a 3-dimensional "soundfield" onto a standard mixing board, a problem that is currently being grappled with by the Virtual Reality industry, is therefore entirely removed. Our goal now is to determine what kinds of tools would be most appropriate for this environment and then design a haptic display device to implement these tools. In so doing we will have designed a system for sound editing that once more relies on physical interaction with the sound recording or representation material. This process will inevitably give rise to a system which does not rely on computer graphics and which is therefore again accessible to the blind.

7 Conclusion

Through this project, we intend to create new audio engineering tools which will begin to blur the line between what is defined as a musical instrument and what is defined as an audio editing tool. At present, audio engineering tools are hard to think of as musical instruments and instruments are not generally thought of as tools. Somewhere in between, there are a host of sound manipulation devices waiting to be designed and exploited in creative work. In particular, we believe the open area for exploration is in the field of haptic interface. It is through mechanical contact that relationships to tools and instruments are developed, muscle memory used, and internal models of the musical or recording processes built up. The internal intuitive models which are developed for manipulation of musical instruments and the rather abstract and analytical models which are applied for sound editing deserve to be intertwined.

Beyond exploring the metaphor of the instrument, we want to explore metaphors offered by the recording technologies themselves. Essentially, we expect that interfaces to our virtual worlds which are heavily influenced by hard real-world designs will be most successful. The waveform is a starting point as is the tape/tape head, and the record/stylus. Some aspects of these paradigms are not mechanically accessible, others are. For the first time, we are in a position to combine, modify, and extend aspects from each of these paradigms and create superior audio engineering tools.

References

with Turbo C++. SAMS Macmillan, Carmel, Indiana, 1991.

[Kelly, Salcudean 1994] Kelly, A.J., and Salcudean, S.E. *On the development of a force-feedback mouse and its integration into a graphical user interface* in Proceedings ASME WAM, Dynamic Systems and Control, DSC-Vol. 55-1, 1994.

[Akamatsu, Sato 1994] Akamatsu, M., Sato, S. and MacKenzie, I.S. *Multimodal mouse: A mouse-type device with tactile and force display* Presence, Vol 3 No. 1, Winter 1994 pp 73-80.

[TU-Berlin] *Haptic Trackball* Personal Communication

[Linvill 1973] Linvill, J. *Research and development of tactile facsimile reading aid for the blind: the optacon* [Washington] U.S. Dept. of Health, Education and Welfare, Office of Education, Bureau of Education for the Handicapped, 1973.

[Chafe 1986] Chafe, C. and Jaffe, D. *Source separation and note identification in polyphonic music* IEEE Conference on Acousics, Speech, and Signal Proccesing, Vol 2, 1986, pp. 1289-1292.

[Loubal 1992] Loubal, P.S. *Fingertip maptracing devices for the blind.* Proceedings of Conference. Technology and Persons with Disabilities. Held: Los Angeles, CA, USA, 18-21 March 1992. 1992. p. 315-18

[Wiker 1991] Wiker, S.F., et al. *Development of tactile mice for blind access to computers: importance of stimulation locus, object size, and vibrotactile display resolution.* Proceedings of the human factors society 35th annual meeting. vol.1 Held: San Francisco, CA, USA, 2-6 Sept. 1991. p. 708-12

[York 1989] York, B.W., et al. *Tools to support blind programmers.* 17th Annual ACM Computer Science Conference. Held: Louisville, KY, USA, 21-23 Feb. 1989. p. 5-11

[Edwards 1989] Edwards, A.D.N. *Soundtrack: an auditory interface for blind users.* Human Computer Interactions vol.4, no.1, 1989 p. 45-66.

[Frisken-Gibson 1987] Frisken-Gibson, S.F., et al. *A 64-solenoid, four-level fingertip search display for the blind.* IEEE TRANSACTIONS

[Eberhardt 1993] Eberhardt, S.P., et al. *OMAR a haptic display for speech perception by deaf and deaf-blind individuals.* Proceedings of IEEE Virtual Reality Annual International Symposium. Held: Seattle, WA, USA, 18-22 Sept. 1993. p. 195-201

[Minsky 1995] Minsky, M. *Computational Haptics: Texture* PhD Thesis, MIT Media Labs, 1995.

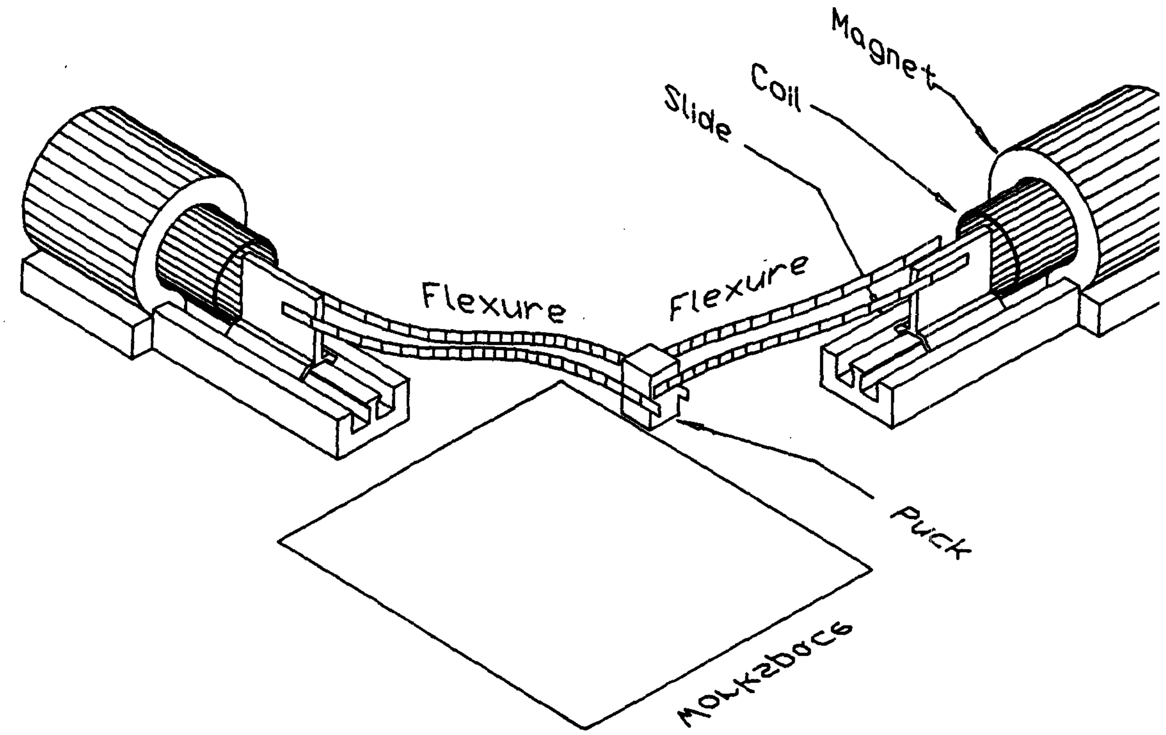
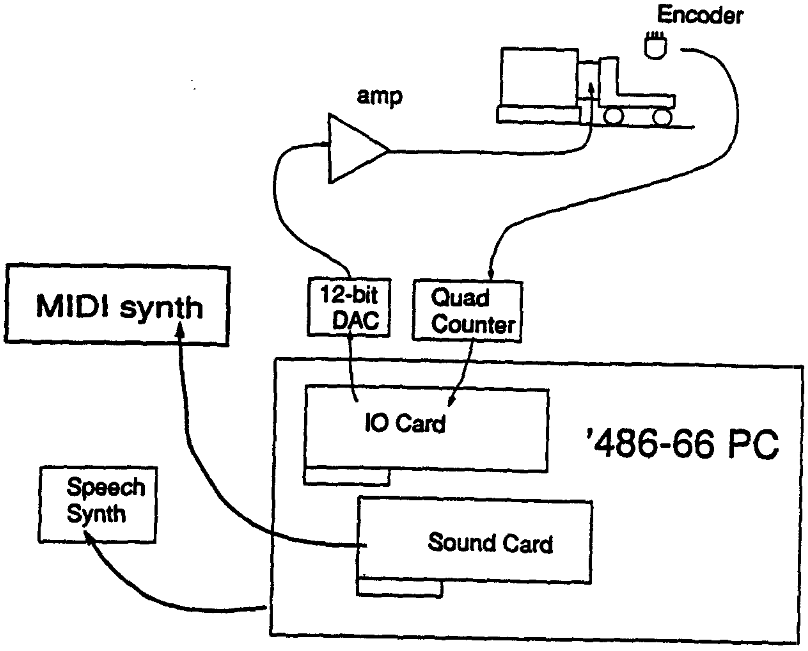


Figure 1. The Moose



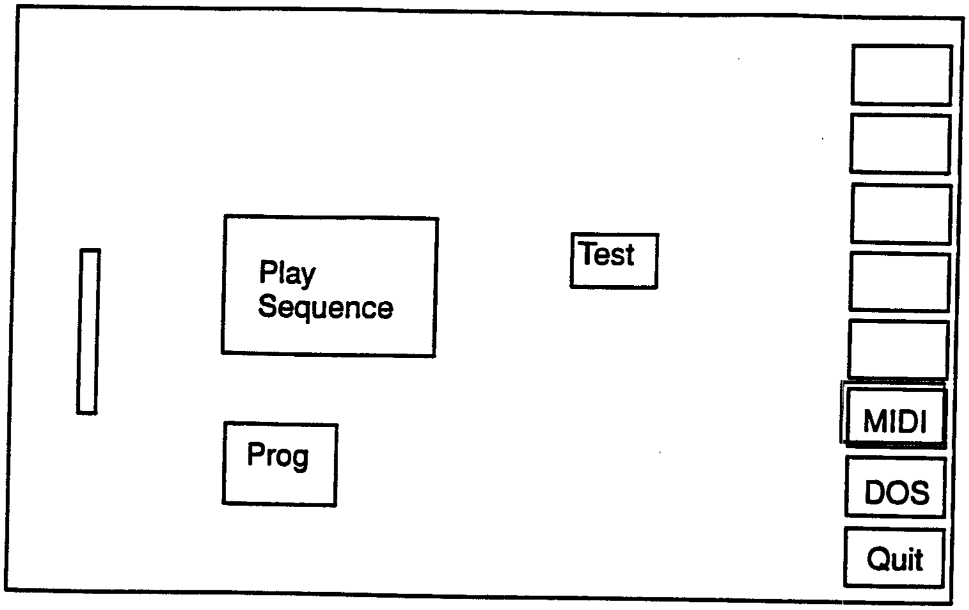


Figure 3. The MIDI Page

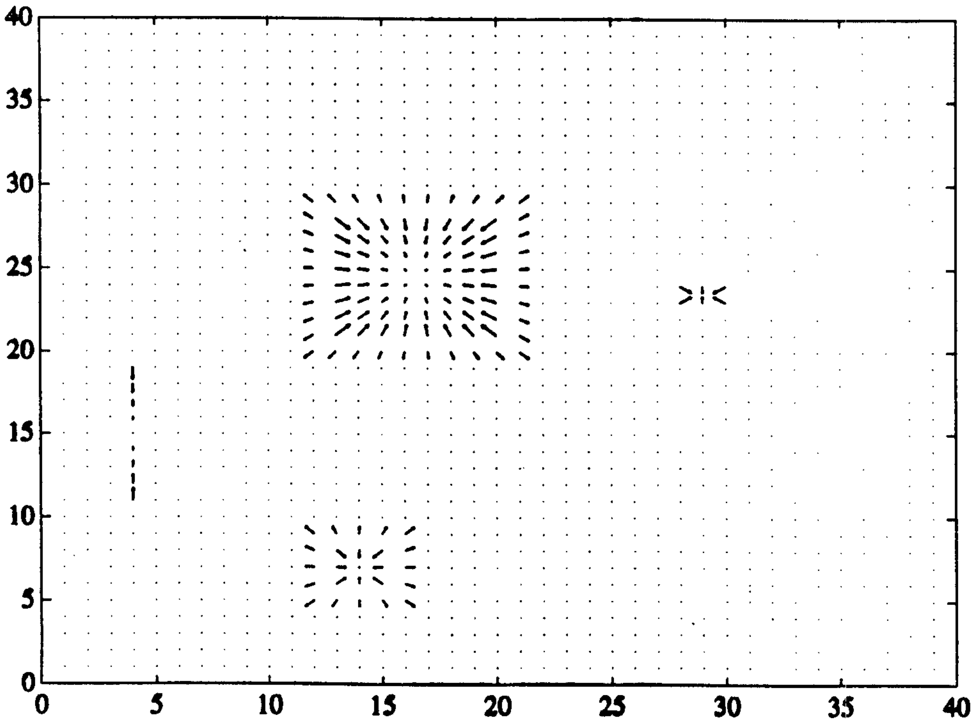


Figure 4. A Typical Force Field

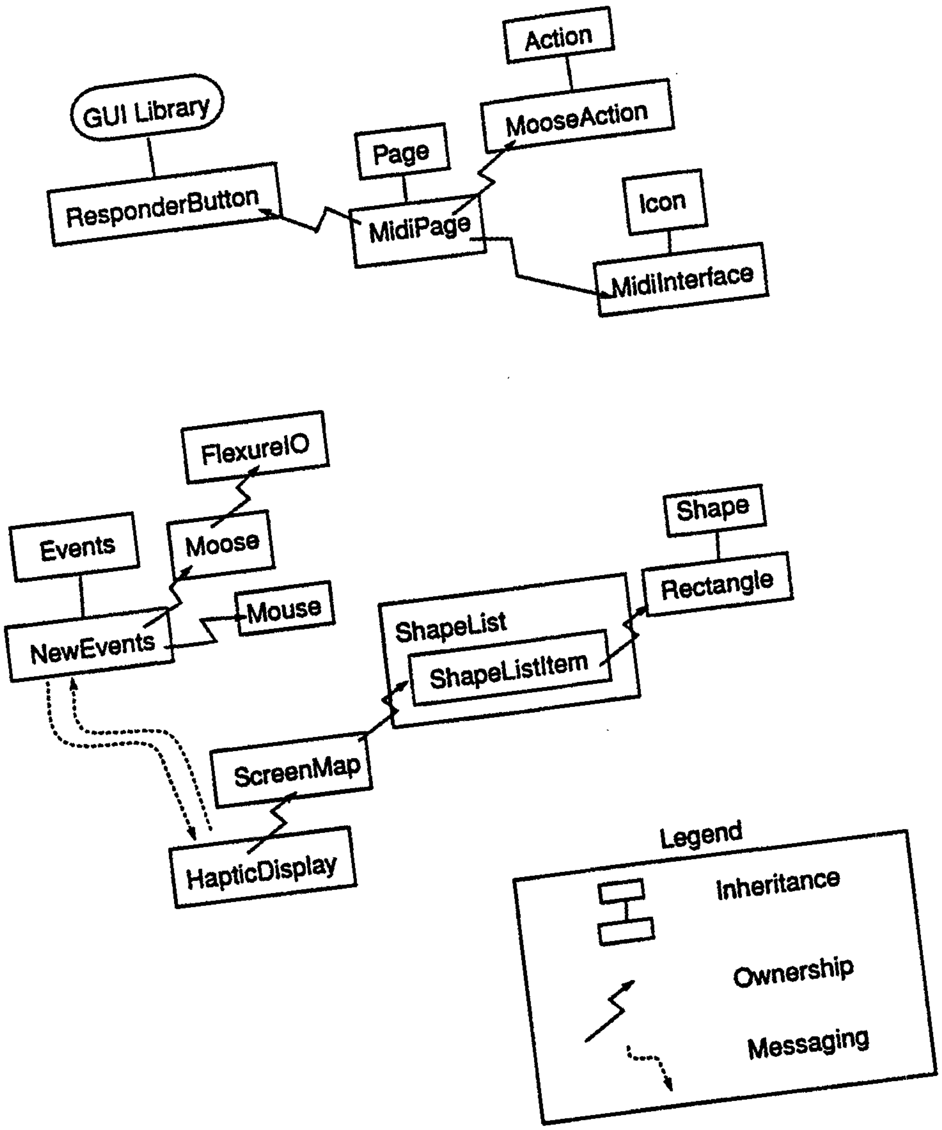


Figure 5. Software Architecture

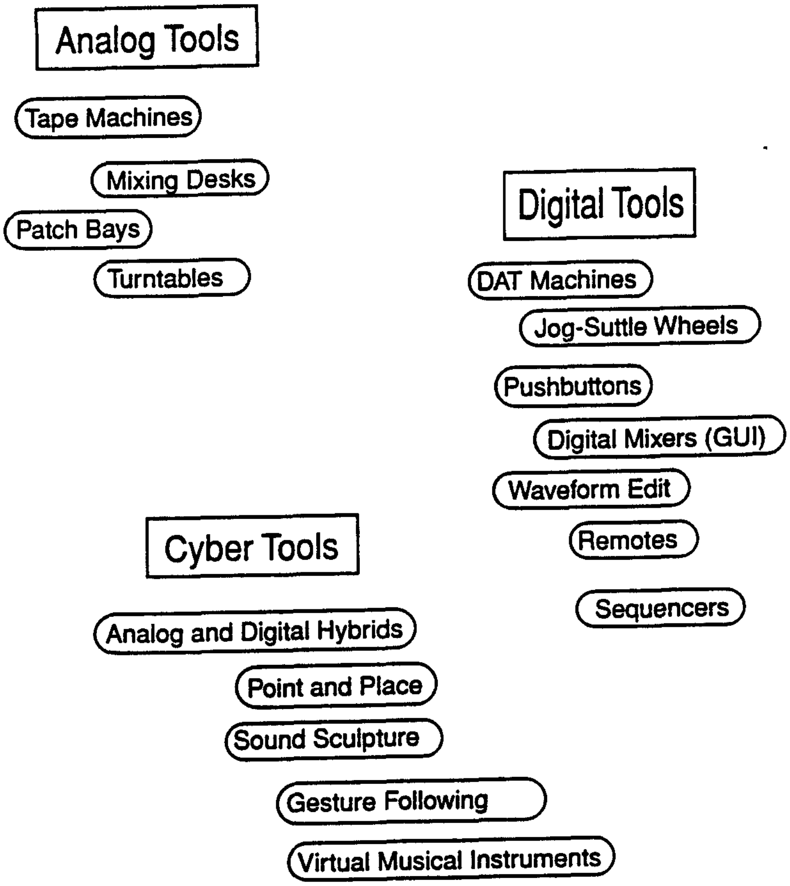


Figure 6. A Taxonomy of Sound Tools