THE DEVELOPMENT OF A SONICALLY-ENHANCED WIDGET SET

Stephen A. Brewster

VTT Information Technology,
Tekniikantie 4 B,
P.O. Box 1203,
FIN-02044 VTT, Finland
Tel.: +358 0 456 4311
stephen.brewster@vtt.fi

INTRODUCTION

Several experiments have shown that the addition of non-speech sound to graphical human-computer interfaces can increase usability [3, 6, 7, 8]. However, because this area is still in its infancy, sounds are often added in *ad hoc* ways by individual designers and this can lead to them being ineffective, for example Barfield, Rosenberg & Levasseur and Portigal [1, 9]. The aim of the research described here is to help designers who are not skilled in sound design to create effective sonically-enhanced interfaces.

This poster describes a project to develop a sonically-enhanced, multimodal widget set that provides for the use of sound in an effective and consistent way across the human-computer interface. Each interface widget will be analysed to find usability problems and sound added to overcome these. Sound can be used to replace graphical feedback with more a effective auditory form or enhance the existing graphical feedback. Once each widget has been sonified they can be combined into a complete widget set. This widget set will allow a designer to create an interface that makes effective use of sound.

Why should sound be added to human-computer interfaces? The combination of graphical and auditory information at the interface is a natural step forward. In everyday life hearing and vision combine to give complementary information about the world. The advantages offered by the different senses can be brought to the computer interface. At the present time almost all information presented by computers is visual but a multimodal interface that integrated information output to both senses could capitalise on the interdependence between them and present information in the most effective way possible.

SOUNDS USED

Earcons [2] will be used to sonify the toolkit. These are abstract, synthetic sounds that can be used in structured combinations to create auditory messages. They have been shown to be effective at communicating information in sound [3]. The sounds will be controlled by MIDI. Almost all modern computer systems either have built-in MIDI-controlled synthesisers (for example on sound cards or via DSP chips) or can easily be connected to them. Using MIDI will also provide an easy way for users to customise the sounds. Standard synthesiser control software can be used to change the timbre or intensity of the sounds in any widget.

OVERALL STRUCTURE

Each of the widgets in a standard widget set will be sonified. The overall structure of the sounds will be as follows: Each application will have its own timbre and spatial location (via stereo) as a base for all of its sounds. All widgets within an application will use these and modify them by changing the rhythm, pitch, intensity, type of notes used (for example chords) or effects (for example chorus or filtering). Figure 1 shows such a hierarchy. At level 1, the three applications all have different timbres and spatial locations. These are inherited by level 2 and modified with pitch, rhythm, etc. These modifications are constant across applications so that widgets in different applications sound consistent (similar to graphical widgets that look consistent across applications). It is hoped that after using the system, the user would come to associate a certain timbre with a particular application.

In practice, it may not be possible to give all applications a different timbre and spatial location (synthesisers usually only have 127 different timbres and not all of these are distinct). Instead, applications could be grouped (for example, communications or graphics) and each group given a different timbre and location.

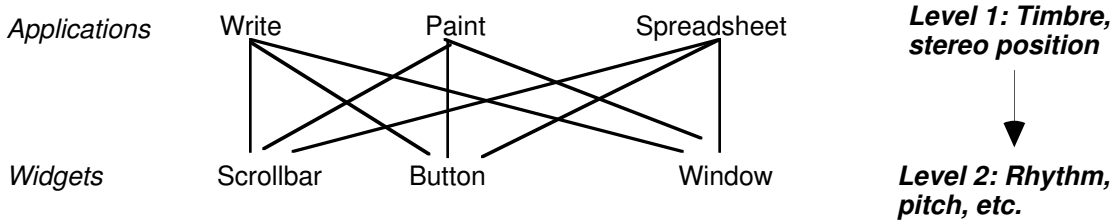

***Figure 1**: A hierarchy of sounds across applications.*

SONICALLY-ENHANCED WIDGETS

There are many different ways that sound could be added to widgets. Here, it will be used to overcome usability problems. Each of the widgets in a standard widget-set will be modelled in terms of its information content. Brewster proposed a structured method that can be used to analyse interactions in terms of event, status and mode information to find where usability problems can occur due to incorrect feedback. From this analysis earcons can be created for the auditory feedback by using existing guidelines (see Brewster [3] for a full discussion). Care will be taken with the design of the sounds so that the widgets can be combined into a whole and still sound pleasant.

Several widgets have already been analysed to find out if there are usability problems that sound could overcome [3]. So far, sonically-enhanced scrollbars, buttons and windows have been experimentally tested to discover if the sounds added were effective at improving usability. When the rest of the widgets have been tested the toolkit will be built up from the individually sonified widgets. This way ensures that each of the widgets has been sonified in an effective way. When testing sonically-enhanced widgets time, error rates, workload, preference and annoyance are measured. This mixture of quantitative and qualitative measures gives a balanced test of usability. A brief description of the sounds to be used for some of the widgets will now be given.

Buttons

One problem that can occur with graphical (on-screen) buttons is that, when trying to press them, users can 'slip-off' by mistake. This happens because they move the mouse off the graphical button before releasing the mouse. This means that the button is not pressed and the action requested does not take place (see [4] for more detail). The feedback from such a slip-off is identical to that of the correct button press so the user finds it hard to tell that it has happened. An analysis was carried out and it suggested where sound could be used.

An earcon was played when the mouse was moved over a button (a quiet background sound). This indicated to users when they were over a target. Another earcon was designed to be the auditory equivalent of the graphical highlight for when the mouse was pressed down on the graphical button. The final earcon indicated when a

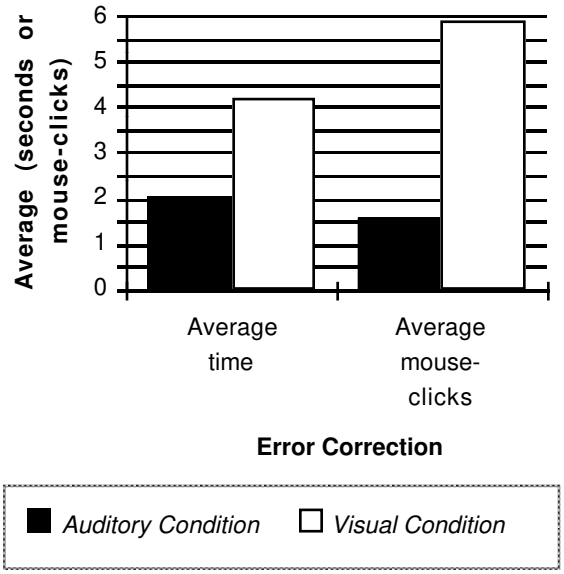

***Figure 2:** Error recovery in standard and sonically-enhanced buttons. The graph shows average error recovery times and the average number of mouse clicks needed for recovery.*

graphical button was pressed correctly or when a slip-off occurred.

Using sound in this way was experimentally tested and shown to be effective. For example, time and the number of mouse-clicks required to recover from slip-off errors were significantly reduced (see Figure 2). In fact the time to recover from a slip-off error was halved [4].

Scrollbars

Scrollbars have also been sonified to overcome scrolling problems. A continuous earcon was added to indicate where the user was in the document. A high pitched sound meant nearer the top and a low pitched sound nearer the bottom. The sound changed pitch when the user scrolled over a page boundary. The constant sound was also modified when the thumb-wheel was dragged. It changed in pitch when the user crossed a page boundary (as before). If the user dragged outside the 'hot-spot' surrounding the scrollbar the sounds stopped. An earcon was also added for scrolling by clicking in the scrollbar. A short, high pitched sound indicated scrolling up and a low pitched sound indicated down.

An experiment was conducted to test a sonically-enhanced scrollbar. It showed that the sounds reduced the workload of the tasks set, they reduced the time taken to complete the tasks and the time taken to recover from errors [5].

Windows

Sounds were added to windows to stop 'unselected window errors', where the user interacts with the wrong window by mistake. Each window was given an earcon with a different timbre and spatial location. A continuous, low-intensity background sound was played when the window of a given application was active to enhance the visual feedback already present. A demanding sound was used to indicate the change from one window to another. The user could tell by the new timbre and spatial location that the application had changed.

Menus and alert boxes

These two widgets have been analysed but not tested. Sound will be utilised to stop users slipping off menu items by mistake and not noticing (a similar problem to buttons). Earcons will be added so that as the user moves up or down the menu a sound will play; a high pitch sound at the top of the menu and a lower one at the bottom, the pitch changing as the user moves from one item to the next. When the mouse button is released over a menu item an earcon will indicate whether the menu item was selected correctly or a slip-off occurred.

A constant earcon will be played when an alert box is displayed. This will inform the user that an important message must be dealt with. The earcon can be manipulated to indicate the level of urgency of the alert (notification, warning or error) by varying the parameters of the sound used. When the alert has been dealt with the continuous sound will stop.

PROGRESS

As mentioned, buttons, scrollbars and windows have so far been sonified and the experimental results are very promising. Menus, dialogue boxes and button palettes have been analysed but have not yet been tested. When they have been evaluated the different widgets can be combined into a complete widget set. One of the strengths of this approach is that each of the sonified widgets will be shown to be effective. The whole widget set will be built on strong foundations. In the final stage of the project, the widget set will be produced. A programmer will be able to use this in the same way as a graphical widget set.

CONCLUSIONS

This poster has described a sonically-enhanced widget set that is under development. Initial investigations have shown significant advantages from adding sound to graphical widgets and a standard sonified widget set will make it easier for designers to created sonically-enhanced human-computer interfaces. Previously, designers had to add sounds to their interfaces in an *ad hoc* way and this lead to ineffective uses of sound. This widget set will allow them to add effective sounds that are consistent across the multimodal human-computer interface.

ACKNOWLEDGEMENT

This work was supported by the ERCIM fellowship programme contract number 94-04.

REFERENCES

1. Barfield, W., Rosenberg, C. & Levasseur, G. (1991). The use of icons, earcons and commands in the design of an online hierarchical menu. *IEEE Transactions on Professional Communication*, 34(2), pp. 101-108.

2. Blattner, M., Sumikawa, D. & Greenberg, R. (1989). Earcons and icons: Their structure and common design principles. *Human Computer Interaction*, 4(1), pp. 11-44.

3. Brewster, S.A. (1994) *Providing a structured method for integrating non-speech audio into human-computer interfaces*. PhD Thesis, University of York.

4. Brewster, S.A., Wright, P.C., Dix, A.J. & Edwards, A.D.N. (1994). The sonic enhancement of graphical buttons. *Submitted to Interact'95*, Lillehammer, Norway.

5. Brewster, S.A., Wright, P.C. & Edwards, A.D.N. (1994). The design and evaluation of an auditory-enhanced scrollbar. In B. Adelson, S. Dumais, & J. Olson (Eds.), *Proceedings of CHI'94*, Boston, Massachusetts: ACM Press, Addison-Wesley, pp. 173-179.

6. Brown, M.L., Newsome, S.L. & Glinert, E.P. (1989). An experiment into the use of auditory cues to reduce visual workload. In *Proceedings of CHI'89*, Austin, Texas: ACM Press, Addison-Wesley, pp. 339-346.

7. Gaver, W., Smith, R. & O'Shea, T. (1991). Effective sounds in complex systems: The ARKola simulation. In S. Robertson, G. Olson, & J. Olson (Eds.), *Proceedings of CHI'91*, New Orleans: ACM Press, Addison-Wesley, pp. 85-90.

8. Perrott, D., Sadralobadi, T., Saberi, K. & Strybel, T. (1991). Aurally aided visual search in the central visual field: Effects of visual load and visual enhancement of the target. *Human Factors*, 33(4), pp. 389-400.

9. Portigal, S. (1994) *Auralization of document structure*. MSc. Thesis, The University of Guelph.