SONICALLY-ENHANCED GRAPHICAL WIDGETS

Stephen A. Brewster
Glasgow Interactive Systems Group
Department of Computing Science, University of Glasgow, Glasgow, G12 8QQ, UK
Tel: +44 (0)141 330 4966, Fax: +44 (0)141 330 4913
Email: stephen@dcs.gla.ac.uk, Web: www.dcs.gla.ac.uk/~stephen/

INTRODUCTION

This demonstration will show many different widgets that have had their usability problems corrected by the addition of non-speech sound. A demonstration is the appropriate place to do this as widgets with sound cannot easily be described in a paper session.

One problem with modern graphical displays is that they are very visually demanding; all information is presented graphically. This can cause users to become overloaded and to miss important information [3]. One reason is that our eyes cannot do everything. Our visual sense has a small area of high acuity. In highly complex graphical displays users must concentrate on one part of the display to perceive the graphical feedback, so that feedback from another part may be missed as it is outside the area of visual focus [4]. This problem is worse for partially sighted users whose area of acuity may be reduced by problems such as tunnel vision.

As an example, imagine you are working on your computer creating a drawing and are also monitoring several on–going tasks such as a compilation, a print job and downloading files over the Internet. The drawing task will take up all of your visual attention because you must concentrate on what you are doing. In order to check when your printout is done, the compilation has finished or the files have downloaded you must move your visual attention away from your picture and look at these other tasks. This causes the interface to intrude into the task you are trying to perform. It is suggested here that some information should be presented in non-speech sound. This would allow you to continue looking at your drawing but to hear information on the other tasks that would otherwise not be seen (or would not be seen unless you moved your visual attention away from the area of interest, so interrupting the task you are trying to perform). Sound and graphics can be used together to exploit the advantages of each. In the above example, you could be looking at the drawing you are creating but hear progress information on the other tasks in sound. To find out how the file download was progressing you could just listen to the download sound without moving your visual attention from the drawing task.

Current interfaces depend heavily on graphical output. One reason for this is that when current widgets (interaction techniques such as buttons, scrollbars, etc.) were developed, visual output was the only communication medium available. However, technology has progressed and now almost all computer manufacturers include sophisticated sound hardware in their systems. This hardware is unused in our daily interactions with our machines (the sounds are really only used to any extent in computer games). This demonstration will show how to take advantage of this available hardware and make it a central part of users' everyday interactions to improve usability.

Even though sound has benefits to offer it is not clear how best to use it in combination with graphical output. The use of sound in computer displays is still in its infancy, there is little research to show the best ways of combining these different media [1]. This means sounds are sometimes added in *ad hoc* and ineffective ways by individual designers. This demonstration will show how sound can be effectively added to current graphical widgets.

Sounds used in the widgets

The sounds used were based around structured non-speech audio messages called *Earcons* [2, 5]. Earcons are abstract, musical tones that can be used in structured combinations to create sound messages to represent parts of an interface. The sounds were created using the earcon guidelines proposed by Brewster *et al.* [6].

The earcons used in the sonically-enhanced widgets were controlled by MIDI. The widgets all use the General MIDI standard so that they can be played on any synthesiser that supports the standard. The widgets send data in real-time to the synthesiser. MIDI data is compact so the computational overhead is low. This means that even slow machines could use the widgets with an external synthesiser attached. Using MIDI also provides a way for users to customise the sounds. Standard synthesiser control software can be used to change the timbre or intensity of the sounds in a widget.

The sonic-enhancements were implemented on top of standard Macintosh graphical widgets. The widgets had code added to them to send MIDI data as interactions were taking place. The current versions of the widgets have the MIDI data to play the sounds hard-coded into them. This is inflexible but allowed them to be tested to assess the effectiveness of the sonic-enhancements. Future versions will have the MIDI data stored as resources. This will allow the sounds to be changed by simply editing a resource file and will enable the use of different *audio themes.* Different themes would be provided (for example there might be a rock or a blues theme) in which all of the sounds for the different widgets would be designed as a coherent set by the interface designer.

The overall structure of the earcons in the widgets is as follows: Each application has its own timbre as a base for all of its sounds (just as an application has a base frame for all of its widgets). All widgets within an application use this and then modify it by changing the rhythm, pitch, etc. These modifications are constant across applications so that widgets in different applications sound consistent (similar to graphical widgets that look consistent across applications). This approach to allocating earcons is the same as for graphical widgets: A graphical application has a base frame for its widgets. This frame has a spatial location and any widget drawn within the frame uses the base spatial location to position itself.

THE SONICALLY-ENHANCED WIDGETS

Several sonically-enhanced widgets will be demonstrated. These will include scrollbars, menus, buttons and tool palettes. This will allow more time for the tool palettes to be demonstrated than will be possible in the paper session where they will be presented. Conference attendees will also be able to try out the widgets themselves. The widgets were designed to solve problems of sighted and partially-sighted users but not completely blind users.

All of the widgets demonstrated have been experimentally tested to ensure that they have improved usability. Results have shown that, with the addition of sound, the time to complete tasks can be reduced, time taken to recover from errors can be reduced and the total number of errors made can be reduced. Qualitative results have shown increased user preference for the sonically-enhanced widgets and reduced subjective workload when operating the widgets [3, 4]. Importantly, there has also been no increase in annoyance due to the addition of sounds into the widgets.

Scrollbars

There are several usability problems with scrollbars that occur because users must look at the scrollbar to interact with it when they really want to carry on with the task they are trying to perform. One problem is lack of position information - it is impossible to tell where you are in a file unless you look away from what you are doing at a page indicator. Earcons were added to overcome this by playing a sound that gave information about the page the user was on - even if he/she was not looking at the scrollbar. The earcon was based on pitch: Higher pitch indicated a location closer to the top of the file and *vice versa*.

Buttons

Sound was added to graphical, on-screen buttons to overcome problems of mis-hitting buttons and slipping off. When this happens the user thinks the button has been pressed when, in fact, it has not, causing usability errors. It is hard for users to notice that they have not pressed it as there is no feedback indicating a mis-hit. Earcons were used to indicate such a problem to users.

Menus

Menus have similar problems to buttons. Users may slip off the menu item they want to one above or below by mistake. This is often not noticed because the feedback is not seen by users; their visual attention has moved away from the menu and back to the information they are working with. Sound has the advantage that it can indicate such problems without requiring users to look away from their main task, allowing them to recover from such errors more quickly. Earcons were added to indicate when users slipped-off a menu item so that they would notice the error had occurred wherever they were looking.

Tool palettes

As will be described in more detail in my short paper presentation, tool palettes have problems as users do not see which is the currently active tool (the information on the tool palette is outside their area of visual focus).

CONCLUSIONS

Earcons have been used to improve the usability of many different graphical widgets. This demonstration will show these widgets and allow conference participants to interact with them and try them out.

The main problem addressed is that most widgets require users to look at them to use them. In many cases users do not want to look at a widget, they want to look at the task they are trying to perform. This causes the interface to intrude into the task and cause errors. By adding earcons, users can get information from widgets without taking their visual attention away from the main task. This improves the usability by reducing the number of errors made, reducing the time taken to recover from them and reducing workload.

REFERENCES

All references by Brewster are available from http://www.dcs.gla.ac.uk/~stephen/

1. Arons, B. and Mynatt, E. The future of speech and audio in the interface. *SIGCHI Bulletin* 26, 4 (1994), 44-48.
2. Blattner, M., Sumikawa, D. and Greenberg, R. Earcons and icons: Their structure and common design principles. *Human Computer Interaction* 4, 1 (1989), 11-44.
3. Brewster, S.A. Using Non-Speech Sound to Overcome Information Overload. *Displays* 17 (1997), 179-189.
4. Brewster, S.A. and Crease, M.G. Making Menus Musical. In *Proceedings of IFIP Interact'97* (Sydney, Australia) Chapman & Hall, 1997, pp. 389-396.
5. Brewster, S.A., Wright, P.C. and Edwards, A.D.N. An evaluation of earcons for use in auditory human-computer interfaces. In *Proceedings of ACM/IFIP INTERCHI'93* (Amsterdam, Holland) ACM Press, Addison-Wesley, 1993, pp. 222-227.
6. Brewster, S.A., Wright, P.C. and Edwards, A.D.N. Experimentally derived guidelines for the creation of earcons. In *Adjunct Proceedings of BCS HCI'95* (Huddersfield, UK), 1995, pp. 155-159.