GUIDELINES FOR AUDIO-ENHANCEMENT OF GRAPHICAL USER INTERFACE WIDGETS

Joanna Lumsden, Stephen Brewster, Murray Crease & Philip Gray
Department of Computing Science
University of Glasgow
17 Lilybank Gardens, Glasgow, G12 8RZ
{jo,stephen,murray,pdg}@dcs.gla.ac.uk
www.dcs.gla.ac.uk/research/audio_toolkit

ABSTRACT

Audio feedback remains little used in most graphical user interfaces despite its potential to greatly enhance interaction. Not only does sonic enhancement of interfaces permit more natural human-computer communication but it also allows users to employ an appropriate sense to solve a problem rather than having to rely solely on vision. Research shows that designers do not typically know how to use sound effectively; subsequently, their ad hoc use of sound often leads to audio feedback being considered an annoying distraction. Unlike the design of purely graphical user interfaces for which guidelines are common, the audio-enhancement of graphical user interfaces has (until now) been plagued by a lack of suitable guidance. This paper presents a series of empirically substantiated guidelines for the design and use of audio-enhanced graphical user interface widgets.

Keywords

Guidelines, toolkit, earcons, widgets, audio feedback.

1. INTRODUCTION

Although research has shown that the inclusion of sound in user interface design can greatly enhance usability (e.g., [5-7, 9]), audio feedback remains little used in most interfaces. Perhaps the primary reason for this is that designers typically do not know how to use sound effectively [11] with the result that, where used, sounds are often employed in *ad hoc* and ineffective ways. Arons and Mynatt suggest one reason for this [1]: "…the lack of design guidelines that are common for the creation of graphical interfaces has plagued interface designers who want to effectively build on previous research in auditory interfaces". The key aim of our research was to investigate effective use of audio feedback and thereby to generate guidelines to assist developers when designing future user interfaces.

With the intention that the guidelines be made accessible to user interface designers, this paper focuses primarily on the guidelines themselves. It does, however, briefly review the background to the guidelines – namely, the development of a toolkit of audio-enhanced widgets which initially served to investigate and thereafter embody an instantiation of the guidelines.

Proceedings Volume 2 of the
16th British HCI Conference
London, September 2002

2. BACKGROUND TO THE GUIDELINES

We conducted a survey into the use of non-speech sound by interface designers. Illustrating that many designers have little or no knowledge of what sound might offer them, nor indeed how to use it, our survey established a strong need for a set of guidelines for the use of non-speech sound in user interface design. In addition, it identified the importance of providing a toolkit of audio-enhanced interface widgets to support designers who lack the skills to directly implement the guidelines [11].

Our aim was to create widgets where sound is not viewed as a second class citizen but as a full part of the interaction experience. The result is a toolkit of widgets that present novel interaction techniques; certain widgets can be presented entirely audibly.

The development of the toolkit provided a real and practical basis for analysing, researching, and testing the guidelines whilst at the same time generating an end product of value to user interface developers.

Developed in Java™ and based closely on the Swing™ widget API, widgets from our toolkit can easily be integrated in any Swing™-based user interface. Tested with several software designers, the toolkit has been found to be robust enough for real world use (it is freely available at www.dcs.gla.ac.uk/research/audio_toolkit). For more

information about the toolkit and the widgets included within it – together with their evaluation – see [7, 9, 10, 12] or visit the web site.

3. GUIDELINES FOR THE USE OF NON-SPEECH SOUND

On the basis of our research, and substantiated by our experimental analysis, we have developed a set of guidelines for the use of audio feedback in graphical user interfaces. These guidelines provide practical advice on how audio feedback (earcons) should be designed for, and used within, the $2^1/_2$D graphical human-computer interface. Fully documented, the guidelines also support the use and extension of the aforementioned toolkit.

The remainder of this paper outlines the non toolkit-specific guidelines, discussed according to the categories into which they are divided (for full details see [10]).

3.1 High-level v. low-level design goals

Guidelines for the use of sound must take into account, and thereby support, the achievement of both high- and low- level design goals. Low-level goals are concerned with the design of individual *earcons* used to present audio feedback for a given widget – i.e. they identify the rôle of individual earcons in an audio-visual interface. A collection of earcons – each annotating a particular widget – is said to form a *sound suite*. Just as a visual style of graphical user interface components (including colour schemes, fonts, border widths etc.) should be consistent across a user interface, so too should earcons in a successful sound suite be consistent in terms of audio style. High-level goals are concerned with the characteristics that a sound suite of earcons should possess.

3.1.1 Low-level design goals – Earcons

Earcon design is concerned with the generation of audio representations for two interface primitives - *widgets* and the *events* widgets generate. Essentially, widgets are interaction techniques which include components such as buttons and interface functionality such as drag-and-drop. Each widget has a distinct, distinguishing visual appearance and position within a graphical display which establishes a context for interaction with that widget. Audio-enhancement of a widget's appearance can extend the usability of the widget either by reinforcing context or, where users' visual attention is not focused on the widget, by wholly communicating context. Events are interface messages that communicate action or state. *Action* events communicate the immediate result of users' interactions with widgets and are associated with discrete foreground tasks that require some degree of visual attention; in these cases, audio cues are added to ensure that all of a widget's events are communicated successfully, effectively, and unambiguously without overloading the visual channel. *State* (or *status*) events communicate the progress of an ongoing task and are primarily associated with background activity which could ultimately be monitored audibly except where/if visual attention is required for initiation, acknowledgement, abortion etc. purposes. Together, widgets and events create the 2D information space of the graphical user interface: along the widget axis, auditory and visual cues communicate widget *type*; along the events axis, these cues signal the state of an interaction. These issues must all be taken into account when designing individual earcons to represent this level of feedback.

3.1.2 High-level design goals – Sound Suite

Simply designing earcons for individual widget types and their events does not guarantee a successful user interface. When embedded within the same interface, earcons from different widgets may interfere with each other such that sounds appear disassociated from their source and/or annoy or fatigue the user. It is therefore important to identify the following goals for a sound suite as a whole:

- *Minimise Annoyance:* excess intensity variations and the overall loudness of audio feedback are the main reported causes of audio-related annoyance [4] and should therefore be avoided;
- *Simplify Mapping:* like purely visual interfaces, audio-enhanced interfaces can become cluttered; the result – users disable the sound. It is therefore important to minimise the total number of different earcons within a sound suite. This can be achieved by ensuring that: the overall mapping between sounds and their associated widget is simple and obvious; and that the overall number of concurrently playing sounds is not excessive;
- *Facilitate Segregation:* earcons associated with a particular widget must always be perceived as emanating from that widget. When a user perceives a sequence of earcons as coming from one widget, this forms an elemental association which can speed up user recall. Sounds are perceived as forming coherent groups if they are *similar* and *proximal* [3]. Earcon proximity can be perceived along the time or frequency axes (and to a lesser extent the spatial axes).

3.2 Human perception of sound

The first subset of guidelines is concerned with the way in which human perception of sound relates to the construction of earcons and their subsequent inclusion in interface widgets:

1. Sounds used to identify widgets should be absolutely distinguishable (i.e. without reference to a relative comparison scale). Timbre – being uniquely distinguishable – should therefore be used in preference to pitch or loudness about which humans can only make relative judgements;

2. The characteristics of widgets and sound sources should be carefully mapped so that the auditory feedback requirements of the former can exploit the auditory features of the latter;

3. Rhythmic motives (short melodies that can be recognised as individual entities) should be used to encode events which communicate the value of a time-varying parameter;

4. Earcons should be kept within a narrow intensity range (suggested range is max. 20dB and min. 10dB above background threshold [14]) so that if the user changes the overall volume of the audio output on his/her computer, no one sound will be lost and no one sound will stand out and be annoying;

5. Earcons should be playable at different tempos so that they can keep pace with interaction. Earcon duration can be minimised by: minimising the sound duration of each individual sound component; playing only the beginning and end components of long earcons during rapid user input; and/or playing earcon components in parallel to speed up presentation for experienced users [8];

6. Audio signatures should be distinct; it is better to use different timbre families than to rely on users' intra- timbre-family recognition which is considerably weaker [14];

7. The audio feedback for an individual event should sound like a complete unit; to achieve this, the first note should be accentuated and the last elongated;

8. Since synchronicity between sensory modalities is an important factor contributing to the perceptual binding that exists when one event generates stimuli in several sensory modalities it is important to avoid, when combining the use of visual and auditory stimuli, audio leads greater than 90ms and audio lags greater than 180ms which are considered annoying.

3.3 Designing individual earcons

The following collection of guidelines relate to the design of specific earcons for any given individual widget type:

9. The absence of sound where a sound is expected will only alert a user to a problem if the expected sound would have been generated as the direct result of a user action and not as a piece of background information;

10. The sounds used to represent audio feedback should be ranked in order of importance so that it is possible to play only the most appropriate when resources are limited;

11. Sounds should not only be mapped to events that are directly related to users' interactions but also to changes in the system's data model;

12. The number of different sounds used should be limited by carefully analysing the requirements of the task/interaction rather than naïvely mapping a different sound to each event. Reusing event/audio signatures across widgets minimises the total number of mappings users must learn and re-enforces the meaning of each;

13. Sounds should provide useful information that users cannot adequately obtain from less intrusive sources;

14. Complex earcons should be constructed using instruments and rhythm in an analogous way to music;

15. No more than six notes per second should be used when employing note repetition to convey information since users can find more than six notes (in that time period) difficult to differentiate;

3.4 Combining simultaneous earcons

The final group of guidelines relate to the combined use of audio-enhanced widgets within the same user interface:

16. If several sounds are playing simultaneously, the absence of a sound *may not* be sufficient feedback to alert a user to a problem (this is especially true if, under these conditions, the user has not taken some direct action for which he/she expects audible feedback – see guideline 9);

17. If the feedback from multiple audio-enhanced widgets *may* be played simultaneously, widgets' feedback should be prioritised so that, where necessary, only the most important is played;

18. When combined in one interface, widgets' audio feedback should be modified according to the relative priority of the widgets together with the importance of the individual earcons within each widget;

19. Earcons associated with widgets used to represent foreground tasks and those used to reflect background activity should be moderated so that they do not mask each other and so that the audio feedback for background activity is sufficiently demanding that it will not be missed by the user;

20. Where the graphical representation of background activity may be obscured by that of foreground tasks, the audio feedback for the former should be complete in its representation. Where this is not the case – and at the extreme, the feedback is only represented graphically – users are likely to miss most or all of the background activity [12];

21. Earcons can be spatialised to allow users to differentiate multiple instances of the same widget type – this can prevent the need to modify their audio feedback design when used collectively within the same user interface.

4. CONCLUSIONS

The guidelines presented above are intended to provide assistance to developers wishing to include audio feedback within their user interface designs. Although following these guidelines by no means guarantees a *good* audio-enhanced user interface, it should be possible to avoid the pitfalls commonly associated with audio enhanced user interface design. It is extremely important that any new design – be it a new audio-enhanced widget or the combined use of such widgets – be thoroughly evaluated. Bad audio feedback design is counter productive to the interaction advantages presented by audio-enhanced widgets; it is likely to be perceived as annoying and therefore bias uses against audio-visual user interface design.

Thorough evaluation of audio-enhanced widgets is complex and time-consuming. Widgets should be evaluated in their own right and in combination to observe the effectiveness of audio moderation (see section 3.4). Since evaluation design is unique to the widget or collection of widgets being observed, it is not possible to provide meaningful *general* guidance in this respect. However, the research underlying the guidelines discussed in this paper presents *many* examples of successful evaluation of audio-enhanced widgets and as such we would point readers to the appropriate papers from our publications list (available from www.dcs.gla.ac.uk/research/audio_toolkit).

5. ACKNOWLEDEMENTS

This research was funded by EPSRC grant GR/L79212.

6. REFERENCES

[1] Arons, B. and Mynatt, E. (1994), The Future of Speech and Audio in The Interface, *SIGCHI Bulletin,* **26**(4), pp. 44 – 48

[2] Barfield, W., Rosenberg, C. and 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

[3] Bregman, A.S. (1994), *Auditory Scene Analysis,* MIT Press

[4] Brewster, S.A. (1994), *Providing a Structured Method for Integrating Non-Speech Audio into Human-Computer Interfaces*, Ph.D., Department of Computing Science, University of York, York

[5] Brewster, S.A. (1997), Using Non-Speech Sound to Overcome Information Overload, *Displays,* **17**(Special Issue on Multimedia Displays), pp. 179 - 189

[6] Brewster, S.A. (1998), Using Earcons to Improve the Usability of Tool Palettes, In *Proceedings of ACM CHI'98*, ACM Press, Addison-Wesley, Los Angeles, Ca, pp. 297 - 298

[7] Brewster, S.A. and Crease, M.G. (1999), Correcting Menu Usability Problems With Sound, *Behaviour and Information Technology,* **18**(3), pp. 165 - 177

[8] Brewster, S.A., Wright, P.C. and Edwards, A.D.N. (1997), Parallel Earcons: Reducing the Length of Audio Messages, *International Journal of Human-Computer Studies,* **43**, 153 - 175

[9] Crease, M. and Brewster, S.A. (1999), Scope for Progress - Monitoring Background Tasks With Sound, In *Proceedings of INTERACT'99*, British Computer Society, Edinburgh, UK, pp. 19 - 20

[10] Lumsden, J. and Brewster, S.A. (2001), *Guidelines For Using the Toolkit of Sonically-Enhanced Widgets*, University of Glasgow, Department of Computing Science Technical Report TR - 2001 - 100, October 2001

[11] Lumsden, J. and Brewster, S.A. (2001), *A Survey of Audio-Related Knowledge Amongst Software Engineers Developing Human-Computer Interfaces*, University of Glasgow, Department of Computing Science, Technical Report TR - 2001 - 97, September 2001

[12] Lumsden, J., Wu, A. and Brewster, S.A. (2001), *Evaluating the Combined Use of Audio Toolkit Widgets*, University of Glasgow, Department of Computing Science Technical Report TR - 2001 - 101, October 2001

[13] Portigal, S. and Carey, T. (1994), Auralization of Document Structure, In *Proceedings of ICAD'94*, Santa Fe Institute, Santa Fe Institute, Santa Fe, pp. 45 - 54

[14] Rigas, D.I. and Alty, J.L. (1998), How Can Multimedia Designers Utilise Timbre?, In *Proceedings of CHI'98*, pp. 274 - 28