Making Progress With Sounds - The Design & Evaluation Of An Audio Progress Bar

Murray Crease & Stephen Brewster

Department of Computing Science, University of Glasgow, Glasgow, UK.
Tel.: (+44) 141 339 8855 x8475
email: <murray/stephen>@dcs.gla.ac.uk Web Site: http://www.dcs.gla.ac.uk/~murray

Abstract

This paper describes an experiment to investigate the effectiveness of adding sound to progress bars. Progress bars have usability problems because they present temporal information graphically and if the user wants to keep abreast of this information, he/she must constantly visually scan the progress bar. The addition of sounds to a progress bar allows users to monitor the state of the progress bar without using their visual focus. Non-speech sounds called earcons were used to indicate the current state of the task as well as the completion of the download. Results showed a significant reduction in the time taken to perform the task in the audio condition. The participants were aware of the state of the progress bar without having to remove the visual focus from their foreground task.

1 Introduction

Progress bars are a common feature of most graphical interfaces. They are used to indicate the current state of a task which does not complete instantaneously, such as downloading documents from the Web or copying files from one disk to another. Myers [8] showed that people prefer systems with progress indicators, as they give novices confidence that a task has been accepted and is progressing successfully, whilst expert users can get sufficient information to predict the approximate completion time of the task.

Examples of the two main types of progress bar are shown in Figure 1. One type of progress bar indicates only that the task is being processed without any indication of the length of time it will take to complete the task. This type of progress bar is often used where the application does not know how long the task will take. Figure 1(a) shows two progress bars in Netscape Navigator on the Mac. At the top right of the window is a 'N' with a moving starscape to indicate progress. Along the bottom of the window is a rectangle with diagonal stripes which 'move' along the rectangle as if it were a pole being rotated. Other examples of similar progress bars are busy cursors (e.g. watch face or hour glass cursors) and information put on a screen to distract the user whilst a task is progressing (e.g. product information being displayed whilst the product is being installed).

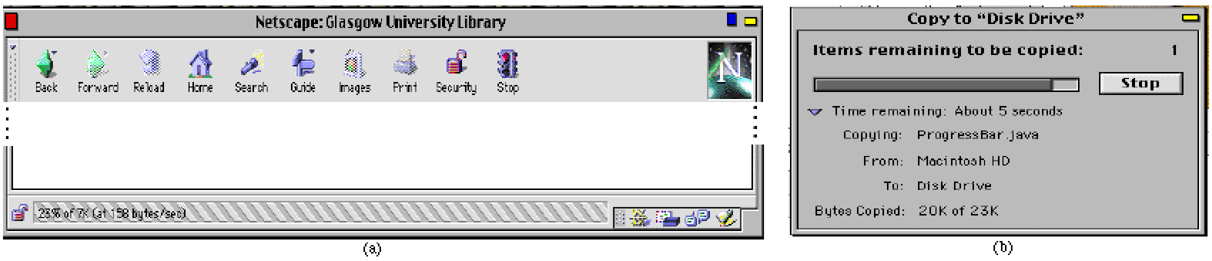

Figure 1 : A Percentage Done Progress Bar

A second type of progress bar, the percentage done progress bar, indicates the amount of the task completed in comparison to the size of the whole task. An example of a percentage done progress bar taken from the Macintosh Finder is shown in Figure 1(b). In this case, a rectangle is filled from left to right in proportion to the amount of the task completed - here a file is being copied from one disk to another. Additionally, textual information is given describing the status of the task in more detail.

2 The Problems With Existing Progress Bars

For a progress bar to be effective at keeping the user informed about the state of the task, Conn [5] says that the progress bar should have good time affordance, i.e. the user must be able to tell "when things are okay and when there are problems, and can generally predict when a task will be completed." In order for the user to have good time affordance, Conn says the progress bar should give an indication of eight task properties:

1. **Acceptance**. What the task is and whether it has been accepted.

2. **Scope**. The overall size of the task and the corresponding time the task is expected to take.
3. **Initiation**. Clear indication that the task has successfully started.
4. **Progress**. Clear indication of the task being carried out, and the rate at which the overall task is approaching completion.
5. **Heartbeat**. Quick indication that the task is still “alive”.
6. **Exception**. Indication if a task has encountered errors.
7. **Remainder**. Indication of how much of the task remains and/or how much time is left before completion.
8. **Completion**. Clear indication of termination of the task and the status at termination.

The progress bar shown in Figure 1(a) provides four of the eight pieces of information required for good time affordance. Acceptance and initiation are shown by the appearance of the progress bar, heartbeat is shown by the movement of the stripes and completion is shown by the disappearance of the progress bar along with the display of a document done message. Status is given in the main browser window. If the download is successful, the appropriate document is displayed, otherwise a short error message is displayed, e.g. URL not found.

When the percentage done progress bar shown in Figure 1(b) is fully expanded, it provides all eight of the pieces of information required for good time affordance. Acceptance is shown by the description of the source and destination of the file(s) being copied. Initiation and heartbeat are given by the textual description of the amount copied. Scope and remainder are given by the estimation of the time remaining to complete the task. Progress and completion are given by the graphical feedback in the form of the rectangle filling up. In this instance, exceptions are handled at the start of the task alerting the user to any prospective problems. For example, if the target disk does not have sufficient space for the file being copied, a dialogue box is displayed alerting the user. If however, the dialogue box is left in its default state only the time remaining, the number of files remaining to be copied and the graphical progress bar are displayed. If the progress bar is pushed to one side or covered by another window, even this information may go unnoticed by the user.

3 Other Auditory Progress Bars

The SonicFinder [7] was based on the standard Macintosh Finder, using auditory icons[6] in addition to the standard graphics. A sound was used to reinforce the graphical feedback given to a user when a file was being copied. The sound used was that of a jug being filled with water. As the task progressed, the sound of the water being poured increased in frequency, analogous to the jug filling up. The sounds used did not give the user any additional information on the copying task, but merely reflected the graphical feedback. This did have the advantage of freeing the user’s visual focus, but was not a complete solution. The sounds only provided information about the initiation, heartbeat and completion of the task. There is no information given about the scope of the task, the amount of the task remaining or the progress of the task. As with the graphical progress bar, some of this information may be inferred over time. For example, if the sounds are increasing in frequency rapidly over a short period of time then the task can be assumed to be progressing rapidly, but unlike the graphical progress bar there is no obviously defined end point. Despite this, Gaver reported that users found the SonicFinder to be a useful and appealing interface; sometimes even bemoaning the lack of sounds when forced to use the standard, silent Finder interface. Unfortunately, no formal user testing was done on the benefits of the SonicFinder so it is impossible to say whether these benefits are tangible.

Albers & Bergman [1] used auditory icons to enhance the usability of the Mosaic Web Browser. Users were given audio cues on the size and type of the file a link pointed to when the cursor was moved over the link. This information allowed a user to gauge the scope of the task without initiating it. The time expected it would take to transfer the file was given by a tick-tock sound played for a short time proportional to the transfer time. The type of file was given by different sounds, e.g. a typewriter sound for a text file; and the size of the file was given by a piano note with the higher the pitch of the note the smaller the size of the file. Once the download was initiated, pops and clicks were played indicating data transfer. If an error occurred, a breaking glass sound was played. As with the SonicFinder, the sounds used were not a complete solution to the problem, giving only the scope, initiation, heartbeat, exception and completion of the task. Because the sound used to indicate the transfer of data were unchanging, no information about the progress or the amount of the task remaining could be inferred. Again, no formal evaluation of the system was undertaken, but the system did show that sounds could be used to create an ubiquitous audio environment providing users with constant low level audio feedback.

4 The Design Of An Audio Progress Bar

Due to the complex nature of progress bars, it was decided to incorporate only some of the sounds which could be used to create an audio progress bar. Thus four sounds were designed to indicate Initiation, Progress, Heartbeat, Remainder and Completion. Earcons [2] were used for these sounds because their structured nature allowed the information required to be conveyed concisely and because it has been shown that earcons can be played in parallel without comprising their meaning [3].

4.1 End Point Sound

This sound was used to indicate the target point of the task. This sound could be considered to be analogous to the right hand side of a standard graphical progress bar which fills up from left to right. This was a single bass guitar note played for 500ms every second during the download and was of a fixed pitch, C_2 (65 Hz). This sound was played as a discrete note every second rather than a continuous note in order to minimise any annoyance. A bass instrument was chosen because this sound is the root upon which the Progress sound is based, in a similar way that a bass line is the root to a melody in a tune.

4.2 Progress Sound

This sound was used to indicate the percentage of the task done. This sound could be considered to be analogous to the right hand side of the portion which is filled in a standard graphical progress bar. This was a single organ note played for 250ms every second during the download, half a second after the end point sound started. The pitch of this note was used to indicate the percentage completed. The pitch of the note starts at C_4 (261 Hz) and as the task progresses this pitch moved towards C_3 (130 Hz) in proportion to the amount of the task completed. The chromatic scale was used for this progression giving 12 steps before the progress sound is played at the same relative pitch as the end point sound. As with the End Point Sound, this sound was a discrete note in order to minimise annoyance. Several instruments were tried for this sound, but the most successful could all be visualised as the instrument which would lead the melody in a tune. This was consistent with the idea of the End Point sound being similar to a bass line. An organ was chosen purely as a matter of preference for the sound, not because it was more effective than any of the other candidates.

4.3 Rate Of Progress Sound

This sound was used to indicate the current rate at which the task was being completed. For example, if a file was being downloaded, the sound would indicate the current download rate in bytes per second. This sound was played at a low volume, and at a low pitch to ensure the sound was not annoying. Each note used a piano instrument with a pitch of C_2 (65 Hz). Each note was 10ms long. At least two notes would be played at regular intervals every second. As the rate of the task increased, more notes would be played, up to a maximum of 12 per second. Although it was a continuous series of notes, because it was played at a low volume and pitch it was not demanding unless the number of notes played changed. i.e. unless the rate of download changed dramatically. A piano instrument was chosen because the attack time of a piano is short so the notes were still recognisable despite their short duration. For the purposes of the experiment described below, the upper limit above which 12 notes were played was set at 50000 bytes per second. This value could perhaps be changed according to the task. For example the rate at which a file is copied from one disk to another is likely to be faster than the rate at which a file is downloaded from the Internet.

4.4 Completion Sound

Once the task is completed, three chords were played. Each chord consists of two notes played for 250ms with a pitch of C_2 (65 Hz) except for the third chord which was played for 500ms. The notes in the chords were played in different instruments; the end point sound instrument and the progress sound instrument. The chords were played immediately one after the other. This sound was played at a slightly higher volume than the other sounds to make it more demanding as it carried a more important piece of information. Chords were played rather than individual notes to make this sound more demanding. Three chords were played to further distinguish this sound from the combination of the Progress and End Point sound which were played twice a second. The final chord was lengthened to indicate completion. This sound assumes that the status upon completion is success, as there could be no other status in the experiment described below. Should the task complete unsuccessfully, a similar but discordant sound could be used to alert the user. Such a sound was successfully used to indicate that a user had slipped onto an adjacent item when making a selection from a pull down menu [4]. This sound could also be used to alert the user should an exception occur.

4.5 Example Sounds

Sound Example One is a recording of the sounds played for a twenty second download which has a steady but slow rate of progress. The progress sound decreases in pitch at a steady rate with only a few rate of completion notes played each second. Sound Example Two is a recording of the sounds played for a twenty second download which again has a steady rate of progress, but the download happens at a far greater rate of bytes per second. As before the progress sound decreases in pitch at a steady rate, but there are many more rate of completion notes played per second. Sound Example Three is a recording of the sounds played for a twenty second download where the rate of download decreases with time. Initially, the progress sounds decrease in pitch rapidly and there are many rate of progress notes played per second. Gradually, the number of rate of progress notes played per second decreases and the progress sound decreases in pitch less rapidly indicating the download is slowing down. Sound Example Four is a recording of the sounds played for a twenty second download where the rate of download increases with time. Initially, the progress sound decreases in pitch slowly and only a few rate of progress sounds are played. Gradually, as the download increases in speed, the number of rate of progress notes played increases and the progress sound decreases in pitch more rapidly.

5 Experimental Design

An experiment was written in Java on a Macintosh PowerPC. The purpose of the experiment was to determine whether the sounds described above were effective. To accomplish this the experimental design had to make the users need to look at a progress bar, but at the same time require their visual focus elsewhere. This is a situation which can often occur in real life situations. For example, a user may download some files from the Internet, and whilst waiting for the downloads to complete the user will perform another task such as word processing. It was decided that the best way of satisfying these experimental requirements was to have the users type in a series of texts whilst downloading a number of files. It was decided that the users should be given poems to type as it was felt that these would engage their attention

The text area the participants used to enter the texts was put on the left hand side of the screen and the progress bar was placed at the top right of the screen. (Figure 2) Below the progress bar was a button labelled Start. When the participant pressed the Start button for the first time, the first download was begun. This marked the start of the experiment and the participant had to start typing in the given texts. The Start button was greyed out until the download had completed, when the participant had to press it again to start the next download. After all the downloads had completed, pressing the Start button had the effect of ending that part of the experiment. A dialogue box was presented to the participant telling them they could stop**.**

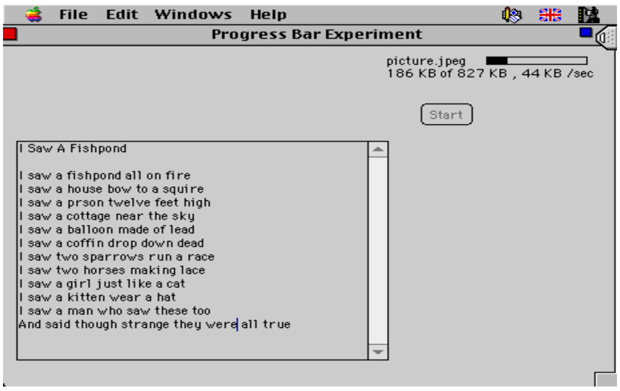

Figure 2 - Experimental Interface

The experiment was a counterbalanced, two condition, within-subjects design. Each participant used both the auditory and standard progress bar. During each condition, the participant had to download 16 files whilst typing in as many texts as they could. This task mimics downloading files as a background task whilst trying to complete a foreground task, in this case typing. Before each condition, the participant was given training in the condition. After each condition the participants were given NASA TLX workload tests to complete, in addition to tests regarding the annoyance the participants felt and their overall preference. These gave a subjective opinion of the changes made to standard graphical percentage done progress bars.

5.1 Experimental Hypotheses

There were three main experimental hypotheses.

1. The overall workload felt by the participants should be less as the participants would find the additional information provided by the sounds useful and the overall preference should increase for the auditory progress bar as the sounds help the participant in the task.
2. There will be no increase in frustration or annoyance due to the sounds as they provide relevant feedback.
3. The time taken to press the button after a download has completed should be reduced in the audio condition as the Completion sound alerts the user more quickly. As a direct consequence of this, the overall time taken for the task should be reduced in the audio condition.

6 Results

The first two hypotheses given above were regarding subjective workload. The overall workload was reduced from 10.25 on average in the visual condition to 8.06 in the audio condition (T_5 = 4.94, p=0.0043) and the Overall Preference was significantly increased from 8.9 in the visual condition to 13.4 in the audio condition (T_{15}=4.29, p=0.0006) confirming the hypotheses. The frustration the participants experienced was significantly reduced to 6.6 in the audio condition from 8.4 in visual condition (T_{15}=2.17, p=0.046) whilst the annoyance did not show any significant difference (7.0 in the audio condition and 8.5 in the visual condition T_{15}=1.48, p=0.159) These figures confirm the hypotheses although the reduction in the frustration experienced was unexpected.

The third hypothesis was regarding objective observations. The time taken to press the button was reduced in the audio condition from 5.3 seconds to 2.8 seconds (T_{15}=5.38, p=0.000008) and the overall time taken for the task was reduced from 726.8 seconds to 687.2 seconds (T_{15}=4.99, p=0.0002), confirming the hypothesis.

7 Conclusions

In this paper we have shown that the addition of sounds can improve the usability of a standard graphical progress bar. The addition of sounds allowed the participants to concentrate on their primary task, typing, whilst monitoring the background task of downloading files. They were thus able to complete the downloading task quicker without compromising their typing task in the audio condition. The sounds used do not fulfil all of Conn's requirements for good time affordance. A fifth sound, similar to the rate of progress sound but played at the start of the download, could be used to give the scope of the task and an indication of acceptance of the task.

Further work needs to be done to discover how these sounds would work in combination, both with other progress bars and with other audio widgets. In the former case perhaps spatialisation of the different multiple progress bars would be sufficient to allow users to distinguish between progress bars. Different timbres could also be used to distinguish different progress bars. Another issue is that of annoyance over a long term download. If a task takes several hours, the sounds may become annoying to the user over time. One mechanism to overcome this would be to fade the sounds out over time if the task is progressing at a steady rate, only fading the sounds back in if the there is a change in the status of the task.

Acknowledgement

This work made possible by the EPSRC grant GR/L 79212.

References

1. Albers, M.C. & Bergman, E. The Audible Web: Auditory Enhancements For Mosaic. In: CHI'95 Conference Companion, ACM Press, Addison Wesley, 1995: pp318-319.

2. Blattner, M., Sumikawa, D. & Greenberg, R. Earcons and Icons: Their Structure And Common Design Principles. Human Computer Interaction 1989;4 : pp 11-44.

3. Brewster, S.A., Wright, P.C. & Edwards, A.D.N. Parallel Earcons: Reducing the Length Of Audio Messages. In: International Journal Of Human-Computer Studies, 1995; 43(2): pp 153-175

4. Brewster, S.A. & Crease, M. Making Menus Musical. In: Proceedings Of Interact'97 (Sydney), Chapman And Hall, pp389-396.

5. Conn, A.P. Time Affordances: The Time Factor in Diagnostic Usability Heuristics. In: Proceedings Of CHI'95 (Denver), ACM Press, Addison Wesley, 1995: pp 186-193.

6. Gaver, W.W. Auditory Icons: Using Sound In Computer Interfaces. In: Human-Computer Interaction, 1986;2 : pp 167-177.

7. Gaver, W.W. The Sonic Finder: An Interface That Uses Auditory Icons. In: Human-Computer Interaction, 1989;4 : pp 67-94.

8. Myers, B.A. The Importance Of Percent-Done Progress Indicators for Computer-Human Interfaces. In: Proceedings Of CHI'85, ACM Press, Addison Wesley, 1995, pp 11-17.