

Visually Supported Design of Auditory User Interfaces

Palle Klante

University of Oldenburg / Department of Computer Science
Uhlhornsweg, 26121 Oldenburg, Germany
Palle.Klante@Informatik.Uni-Oldenburg.De

Abstract

This paper describes the functionality, the used models and an evaluation of the Visual Auditory Interface Design (Visual-AID) prototyping tool which assists the Usability Engineering Process for Auditory User Interfaces (AUI). The process generalizes the experiences made in the INVITE-ZIB-Project, which creates an auditory Web browser for blind users (Donker et al. 2002). The complete user-centered design process defines three main components: guidelines and rules, auditory interaction objects (audIO) and a tool to generate evaluation mock-ups.

1 Introduction

There is a wide range of application possibilities for blind users, but also for sighted users in e.g. mobile scenarios (Shawney & Schmandt, 2000), where only a small or at least no visual display exists as a result of hardware miniaturisation. Future applications are Location Based Services, which nowadays are used with PDAs. The user walks through a museum or a city and dependent on his position he receives information on objects and exhibits in his reach. With current devices the user has to look at a little screen and can't focus on the exhibits. The amount of information is not limited, because the user can actively navigate through the information space. Special attention has to be given on the change between different applications. What happens when the user leaves the museum and get into his car? Actually the interaction objects are changing. The user has to rethink about the used sounds and the behaviour of the objects. A standardization of objects and the use of one specific toolkit of interaction objects for a specific task seems to be useful.

In contrast to the design and the attributes of graphical presentations, there are some differences when designing with acoustic elements. The auditory information processing is primarily time-based and only has a poor resolution in space. A graphical screenshot gives detailed information about the current system status. But when a single sample of peaks is recorded from an audio output, the scene analysis is more difficult. The user has to listen to it for a longer period. These differences and limitations have to be considered in the components of the Usability Engineering Process of AUI.

2 Usability Engineering of Auditory User Interfaces

Usability Engineering supports the idea of design as an intentional and structured process with activity instructions and methods. A detailed description of the final product cannot be given before the actual start of the project. Instead the target should be described via usability goals which should be reached. The product can finally be tested against the completeness of the goals. When designing AUI these design goals have to be adapted and additional criteria have to be considered (e.g. pleasant for a longer period, navigable, easy localization, distinct, aesthetic, joy of use, etc.). These criteria can be categorized in a catalogue of guidelines.

After establishing the usability goals the task for which the application should be build must be identified. The tasks have to be split up in smaller subtasks. A sequence of subtask describes a complete task. These sequences are describing the application flow. To reduce the design complexity early in the design process a disjunction must be done between the model of the user interface and the elements which are used in this model. After finishing the application flow, the elements in each scene must be described. This is partly based on game design. Usually the design team creates a storyboard, then implements the game structure and game idea in one team. Another team implements the figures and actors which can be used in the game structure. It has to be ensure, that every team member of the interdisciplinary design team works with the same design idea. The directions have to be given in a style guide at the beginning of the design process.

For the design of AUIs the scene concept is useful and is the basis for this work. The metaphor of a storyboard is adapted from film production, which also uses time-dependent media and not static elements like in GUIs. The storyboard consists of little stories, which are sequences of scenes. The storyboard is a scene graph. In each scene actors are jumping in, moving in it and finally leaving. Each scene has to be designed for its own, but must also fit to the design of the other scenes.

2.1 Auditory Interaction Objects (audIO)

The design of the actors in a scene are split up in two categories. On the one hand you have to design the information objects. This is the content of the application. The objects which are manipulated by the tools and functions the application offers. The objects are specific to a specific application. A designer has to invent the presentation and behaviour of these objects always from the beginning. It can be described as a picture in an imaging tool. On the other hand there exist interaction objects. The objects are called widgets in GUIs. They offer functions presented in menu entries, buttons to start an action or edit boxes to enter parameters. The audIOs' are based on the concept of abstract interaction objects (Vanderdonckt, 1995). Each object will be described by it's name, attributes, events, behaviour, sound sources and associated interaction objects. The audIOs should present typical tasks, which are often used.

The basis for the sound design of passive information objects as well as active interaction objects depend on the content of the object. The concept of auditory icons (Gaver, 1989) is used for a limited number of objects which convey a certain meaning. But if objects appear in a large and not limited number, such as the representation of documents, the concept of composed Earcons (Blattner et al. 1989) is used. The finally used concept for presenting audIO are Hearcons (Bölke & Gorny, 1995). Hearcons are permanently hearable objects with a position in space, a varying intensity, an interaction area and a specific sound.

Different sets of audIOs are necessary for different application models. This depends on the used metaphor, and the input-and output devices. The control of the interaction is fixed in the dialogue structure which is controlled by the application. An integration of the audIOs in other applications is easily possible.

3 Visual AID

The prototyping tool Visual-AID is used in different steps of the process. It is a platform for design team members to discuss sound design solutions, for building the above mentioned dialogue models and structures and for evaluating complete prototypes as well. It had shown that paper prototypes are not suitable to perceive the difficulties of an auditory representation. You cannot imagine the actual auditory output just by seeing the objects on a visual display. Therefore it is necessary to present changes in the dialogue model of the AUI already in the development

process. The presented tool considers this fact by having an author- and simulation modus. Every single change of the interface definition has a direct impact on the auditory presentation.

In general, GUI-Builders are interaction-based and support a layer-based presentation. AUI are time-based and it is impossible to overlap objects. This is considered by the prototyping tool which consists of four views: The application's core is the **scene and object selection**. It represents the structure of the generated auditory prototype in a tree view and it is technically saved as an XML-document. So it can be exported to other applications and transformed to other user interfaces. First a project must be generated in which different scenes can be integrated. For each scene a main interaction object can be defined. An object can be selected in the icon bar, and will automatically set up in the selected scene or other object. This object will start when the application flow starts the scene. A scene can have only one main interaction object, but this one object can activate various other audIOS within the scene. In the **object property window**, which is connected to the scene and object selection, the attributes for a selected object (the project, interaction space, the scenes and used audIOS with sound sources) can be manipulated. This window is mainly used to define starting conditions for a scene and in what kind of mode it should start.

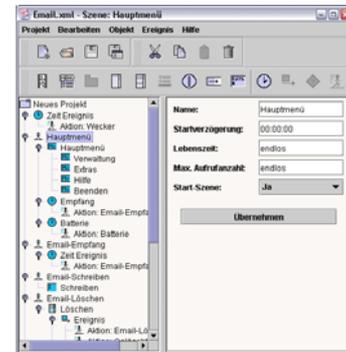


Figure 1: Object selection and property window

The scenes are finally organized in a scene graph. The change from one scene to another is initialised through a user interaction captured by an audIO. The audIO can throw external or internal events. External events have effects on other scenes, while internal events only have effects within the same scene. When an external event is thrown it is possible to define conditional actions. The events will then have different effects. E.g. a scene with the main menu of an application should only be started after a welcome message has finished playing. As well as it is possible to define interaction events from the user, it is also possible to define automatically started time events for a scene. They are used to simulate events from outside the applications. E.g. while selecting E-Mails from a list a message appears after a few seconds

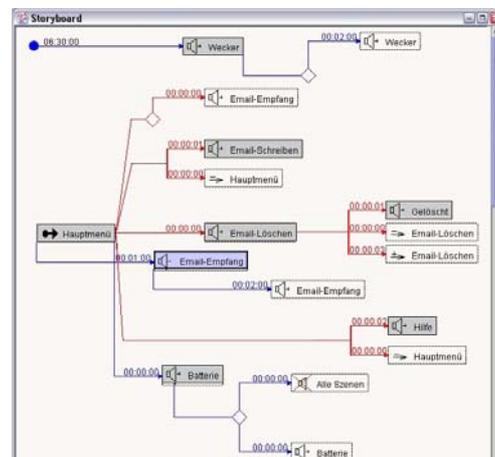


Figure 2: Storyboard and scene graph

to make the user aware, that the battery is low, and the application has to be shut down in a few minutes. The latter scenario shows an important aspect of AUIs: The Designer always has to care about the scenes. Defining a new scene has direct impact of already running scenes. While in GUIs the overlapping or covering of objects has mainly no effects on the correct perception of the actual objects, the audio output will not be recognized correctly from the user, because of interferences. Another problem is considered inside the audIOS. Humans are not able to percept and differentiate more than five to eight auditory signals at once. While the audIOS are never build up with more than five Hearcons, this concept can be broken, when playing more than one scene in the same time. The solution is to change the loudness status for a scene. When the user works in one scene and a dialogue appears in another scene the intensity can be lowered. The underlying

building model of AUI will be visible in the **storyboard window**. A special notation contains vertical lines as time events, diagonal lines as interaction events and a junction for conditional events. The knots are scenes which contain audIOs. Each audIO in a scene can release one of the two event types. The storyboard considers the time-based approach of AUI. By clicking on a scene in the storyboard window, a detailed description of the scene appears. The external events are shown, and a timeline for the time-based events is integrated. Each interaction object that throws an external event, has a direction connection to the resulting scene.

By creating a scene and selecting audIOs for it, the changes will be simultaneously shown in the **visual scene description**. It shows the arrangement of the Hearcons used in one scene. The Hearcon attributes (interaction area, position in space, volume and a specific sound) can be manipulated there. The visual scene description can be used as the application window. Possible interaction from the user can be tested. The author has the possibility to view and hear directly a design decision while he makes them and interact with the audIOs.

3.1 Evaluation

Two evaluation phases were conducted to improve the concept and the usability of the design tool. The results of the first evaluation were already implemented and the description in this paper shows the status of the software at the second evaluation. For both evaluations a set of audIOs' was developed for a mobile scenario. The toolkit contained a Menu, MenuItem, Scrollbar, ProgressBar, Separator, 1ofN Selection, MofN Selection. The interaction objects were navigable with only four input keys and speech input for hands-free interaction and no a visual display.

The first evaluation has its focus on the question what kind of support the user group needs in the design process. It was an evaluation of the concept. In the first evaluation prototype the possibility to define conditional events was not implemented. The support of the audIO was not finished and finally the ad hoc audio presentation in the scene view doesn't work. The result from this first evaluation was to have a direct feedback and more control over the application flow. The second evaluation had to prove that these advanced concept works. The software was reengineered. At both evaluations ten participants took part and only four participants of the first evaluation also were invited at the second evaluation. Four of them were computer science students, who had already experience in creating auditory user interfaces. One participants was developing graphical user interfaces and had lots of experience with visual tools for GUIs. One person mainly generates composed sounds on the computer. There was one musician, which previously has generated some sound for another project, one physicist with the focus of acoustic, and one psychologist who is working in field of human-environment-relation focused on acoustics.

A detailed task description with context-information was defined. The participants should visually implement an auditory navigation software to manipulate the options and navigate in the content of a mobile MP3-Player. The evaluation task was split up into five subtask. It began by creating a main menu, generating a message when the user enters the application. This first tasks were very easy and had the background to show the participants who to use the interaction objects inside the Visual-AID system. The last three tasks were much more complicated. The participants had to use conditional actions and time based events and were ask to simulate a status indicator of a battery, which warns the user every 45sec for a maximum of four times. The participants were asked to think aloud while solving the task. They had to tell what they want to do and what they think where they can find it. The communication with the evaluation leader was rather difficult, because the participants had to hear the output of the system. They must control their design decision and need to know how a scene sounds like. Also the tests were recorded on video. After finishing the test they were interviewed. The whole test last 90 minutes.

All tasks were solved by the participants. One main problem was, that the participants were not involved in the previous concept design and had to edit tasks they did not have invented themselves. The creation of a project was very easy for the participants. The concept was easily understood and internalized. The use of the integrated audIO was also very good. At first it was not obvious (especially for the non computer science group) that the audIO are not part of the application. This is surely a typical behavior of integrated development environments which are mainly used by programmers. So the computer science group had no problem with this integration. The evaluation shows, that there are problems to use the conditional events, as well as the time events. These two features are not normally used everyday and the participants had problems to adapt their ideas to the more formal notation. The participants first have to understand the task and have to think about how to describe it in the software. This was a problem of the lost context. The time events were problematic too, because they can be set on different levels in the prototype. There were global events for the whole applications and local events inside a scene. This in combination with the duration of a scene was hard to manage.

3.2 Conclusion

The direct interaction with the new auditory application was very good accepted. It allows to test the design application ad hoc. It was very easy to test different sound-sources. Mistakes were recognized. It least the main problem during the design process was solved: Not to know, how the final user interface will sound like. The concept evaluation has shown, that the visually supported design of auditory user interfaces helps to test design solution in the design process. Although the prototype user interface is for its own not well designed. There is much work to do, to implement different sets of interaction objects for different usage scenarios. Later it will be necessary to have a connection to the functionality the final application (the user interface prototype is made for) should have. In fact Visual-AID supports a non-formal, flexible and free usage. The tool's architecture takes into consideration to use different sets of independent audIOs'. In the future the tool and the audIO will be made public. Therefore it is necessary to make sure to allow the audio output on a wider sound hardware.

4 References

- Bölke, L. & Gorny, P. (1995) Direkte Manipulation akustischer Objekte. In D.Böcker (HG.), Proceedings Software-Ergonomie'95. Darmstadt 1995. Stuttgart Teubner, p 93-106
- Blattner, M. M.; Sumikawa, D. A.; Greenberg, R. M. (1989), Earcons and Icons: Their Structure and Common Design Principles. In Human-Computer Interaction, Special Issue on Nonspeech Audio; V4; N1. Lawrence Erlbaum, Seite 11-44
- Donker, H.; Klante, P.; Gorny, P. (2002), The Design of Auditory User Interfaces for Blind Users. In Proceedings of NordiCHI 2002, Second Nordic Conference on Human-Computer Interaction October 19-23, 2002, Aarhus, Denmark
- Gaver, W. W. (1989) The SonicFinder. An Interface That Uses Auditory Icons. In Human-Computer Interaction, Spec. Issue on Nonspeech Audio; V4 N1. Lawrence Erlbaum, p 67-94
- Klante, Palle (2002), Werkzeuggestützte Entwicklung auditiver Benutzungsoberflächen, Presented Paper at Informatiktage 2002, Fachwissenschaftlicher Informatik-Kongreß, 8.-9. November 2002, Bad Schussenried, Deutschland
- Sawhney, Nitin; Schmandt, Chris (2000), Nomadic Radio: speech and audio interaction for contextual messaging in nomadic environments. ToCHI: Volume 7, Issue 3 (Sep 2000) Special Issue on human-computer interaction with mobile systems. p 353-383
- Vanderdonckt, J. (1995), Tutorial 12: Tools for Working with Guidelines. HCI International'95, 6th International Conference on Human-Computer Interaction, Yokohama, Japan.