Breemen, A., & Bartneck, C. (2003). An emotional interface for a music gathering application.
Proceedings of the 8th international conference on Intelligent user interfaces, Miami, Florida, USA. | DOI: 10.1145/604045.604107

An Emotional InterFace for a Music Gathering Application

Albert van Breemen
Philips Research, Software Architecture Group
Prof. Holstlaan 4 (WDC-1.034)
5656 AA Eindhoven
The Netherlands
albert.van.breemen@philips.com

Christoph Bartneck
Technical University of Eindhoven
Faculty of Industrial Design
Den Dolech 2, 5600 MB Eindhoven
The Netherlands
christoph@bartneck.de

ABSTRACT
Listening to music while travelling is a pleasant activity. The latest MP3 players demonstrate that storage and management of music will not be a problem in the near future. Besides listening to music the user might also want to gather new music from the Internet. We propose a music gathering application that helps the user to collect music and that is able to proactively search and download music based on the user's music preferences. Furthermore, we developed an emotional interface character that provides instant and natural feedback on the status of the application.

Categories & Subject Descriptors:
J.4 [Computer Applications]: Social and Behavioral Sciences --- Psychology

General Terms: Human Factors, Design

Keywords: Emotions, music, character, agent

INTRODUCTION

Listening to music is a pleasant activity while travelling. Many people would not like to miss it and thus carry Walkmans, portable CD Players or MP3 Players to shorten their journey. Besides listening to music, the user may also use the time on a journey to search for new music and gather it through the Internet. The development of ubiquitous wireless networks makes this feature possible very soon. Some mobile devices, such as the iPaq [3] offer wireless network cards that enable them to use such networks. With this technology the user will be able to access the Internet and gather the desired music.

The music scene develops very fast and new albums are released every day. The user cannot be aware of the latest releases and trends all the time. Therefore the music gathering application should proactively search and download music based on the user's preferences.

The remainder of this article is structured as follows. In the first section we discuss the proactive music gathering application. The user interface design is presented in the second section. Section 3 presents the results we have obtained thus far. Finally, we draw conclusions and discuss future research directions.

PROACTIVE MUSIC GATHERING

Functional Requirements

In order to derive a set of functional requirements we used a scenario-based approach. We developed scenarios of travellers who want to obtain new music. By analyzing the scenarios, we were able to derive a set of functional requirements for our application architecture. The analysis of the scenarios focused on the *information need* of the application to function properly. We identified four needs of information that are essential for realizing proactive music gathering.

First, the application should have information about the *existence of music items* such as specific songs and albums. Second, the application should know what kind of music the user likes and what specific requests the user has with respect to obtaining particular music. The application needs thus a *profile of the user*. Third, the application should have *meta-data about music items*. Meta data is needed in order to reason which music items are liked or disliked by the user. Last, the application needs information about places where to download the music items, e.g. information about *download sites on the Internet.*

Application Architecture

To come to an overall architecture for the music gathering application we've applied several composition principles. We discriminate between non-agent and agent components, as well as between a central agent and support agents. Also, we applied the mirroring of external resources principle. For every relevant information source on the Internet an agent is designed that knows how to access that information.

Figure 1 illustrates the different components of the music gathering application. The non-agent components are the Preference Collection, the MP3 Player and the MP3 Collection. These components are internally structured using traditional software engineering techniques. The agent components are:

- *Music Collector Agent*, a central BDI agent [2;10] who reasons about what music items to obtain.
- *OpenNap Agent*, a support agent that handles the problem of downloading MP3 files from OpenNap servers on the Internet.

IUI'03, January 12–15, 2003, Miami, Florida, USA.

Breemen, A., & Bartneck, C. (2003). An emotional interface for a music gathering application.
Proceedings of the 8th international conference on Intelligent user interfaces, Miami, Florida, USA. | DOI: 10.1145/604045.604107

- *Chart Agent*, a support agent that monitors particular Internet sites with hit chart information. When new chart information becomes available this agent parses the Internet site and sends new hit chart information to the Music Collector Agent.
- *Profile Agent*, a support agent that generates a profile of the user based on information about the user's MP3 collection and on the user's playback behavior.
- *FreeDB Agent*, a support agent that knows how to access the FreeDB Internet site [5] to obtain information about the tracks of an album.

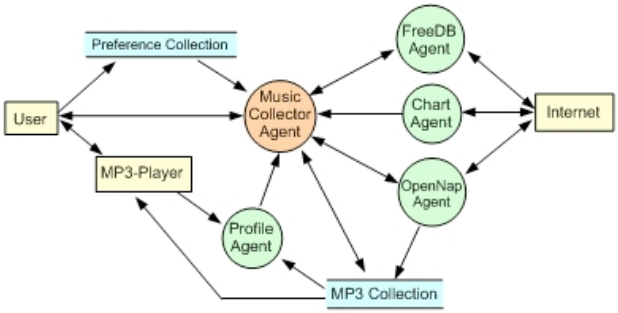

Figure 1: Proactive music gathering application architecture.

Informing the User

The music gathering application is complex and many problems may occur during the execution of the application. However, the user does not need to know all the details about the events that are generated by the application. When the user takes a quick look at his or her MP3 player he or she may only want to roughly know how well the music gathering progresses. If the progress is unsatisfactory the user may want to take actions to resolve the problem. In the next section the design of the user interface is discussed, in particular how it summarizes the information complexity described above.

INTERFACE DESIGN

The design of interface for mobile devices faces several challenges. First, the screen size is small compared to the size of standard computer screens. The popular iPaq, for example has only a 240x320 pixel screen. The screen resolution is just as low as on computer screens which makes it inconvenient to read longer passages of text or to identify small icons.

Second, the mobile context in which these devices are used differs considerably from standard PC workplaces. Mobile devices are usually held in one hand and the other performs the control actions. The devices are also used outside of the user's home or workplace.

Third, one can distinguish the various mobile devices by their input methods. Several devices use a small keyboard or keypad [9], while others use a pen [3;8]. Also combinations of the two main input methods are available [6]. None of the input methods can yet compete with the efficiency and effectiveness of the standard PC mouse and keyboard. Hence, long writing tasks are usually done with help of the PC. However, the control of simple graphical user interfaces (GUI) works rather well on pen-based devices.

Last, the complex information of the status of the music gathering should be available to the user at a glance, similar to checking the time on a watch. Given the small screen size and resolution any of the standard GUI elements, such as multiple progress bars and text logs, seem unsuitable. Therefore, an emotional interface character is used (see Section "Status tab" below). The interface for the music gathering application is optimized to a screen size of 240x320, which is a standard size for current mobile devices. The interface is split into four tabs that correspond with the four steps the user has to perform to gather music.

Search tab

In the search tab the user enters the artist, album or song he or she would like to gather (see Figure 2).

The result of the search is displayed in form of a hierarchical tree structure, ordered by artist, album and song. For example, if the user is looking for music of the band "Galaxy 500" then the result field will display the albums of this band and within these the songs that belong to each album. The user may now select any combination of albums and songs that he or she would like to gather.

Status tab

The status tab provides feedback on the current status of the music gathering (see Figure 2).

The numerous aspects of this status, such as the number of available servers, speed of the download and the availability of chart information are too complex to be visualized given the small screen size. Therefore a comic character face is used to provide a natural and instant feedback to the user. The character is based on eMuu, an embodied emotional character developed by Bartneck [1]. The character uses emotional facial expressions to communicate the status of the gathering application to the user. A simplified OCC emotion model [7] is used to map the numerous events and actions to emotional states and their intensities.

The subsection chosen from the OCC model focuses on the well-being type, creating a character that is able to communicate its internal emotional state to the world. The well-being type emotions are mapped to a set of three different emotional expressions: happiness, anger and sadness. In short, all the positive events and actions will result in happiness (e.g. success download of a song), all the negative events will result in sadness (e.g. an empty search result) and all the negative actions will result in anger (e.g. download abort by peer). The distinction of what is an event and what is an action is based on accountability. It is impossible to blame a person for the failure in the Internet, but if a specific person cancels the download of the user then the character has a person to be angry at. The intensity of each emotional state is based on certain variables such as the likelihood of the event.

Breemen, A., & Bartneck, C. (2003). An emotional interface for a music gathering application.
Proceedings of the 8th international conference on Intelligent user interfaces, Miami, Florida, USA. | DOI: 10.1145/604045.604107

Files tab

The files tab displays the files that are currently in the download directory of the application. All songs downloaded by the user, including the ones being currently processed, are shown in an hierarchy tree. This tree structure allows the user to select any combination of artist, album and songs and to perform actions on the selection. The user may, for example, listen to a song to check its correctness and quality or retry downloading songs that have not been completely downloaded due to an error. Moreover, the user can delete songs of any artist or album or move them to the music library of his or her jukebox application.

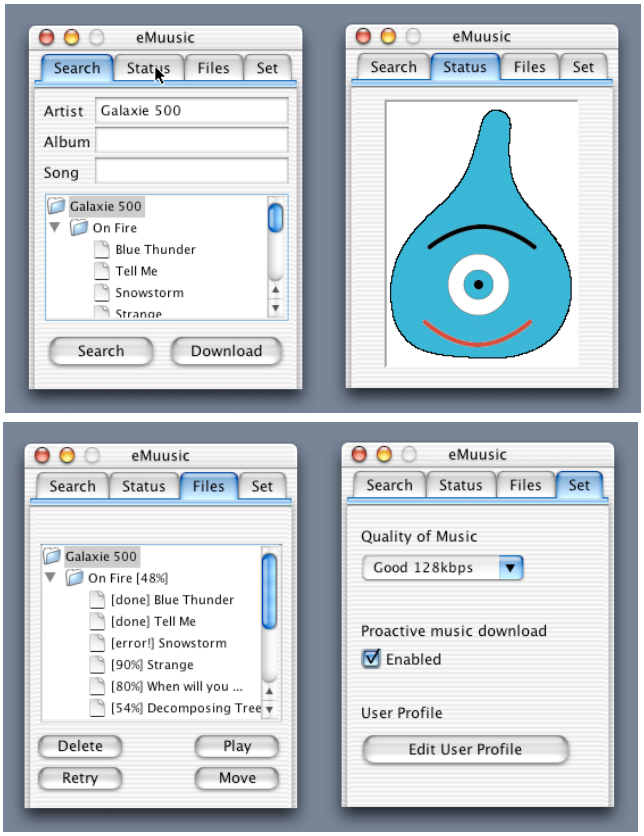

Figure 2: The search, status, files and settings tabs.

Settings tab

In the settings tab the user can adjust the system preferences. The proactive music gathering can be switched on an off, the user's music profile can be edited and the desired music quality for the downloaded songs can be selected from a predefined list.

EVALUATION

Several tests were performed to evaluate the application's performance. The performance was highly satisfactory. The application was able to combine the user's profile with real time hit chart information from the Internet to search and download songs from OpenNap servers. Because the OpenNap Agent incorporated reinforcement learning techniques, the performance of connecting to servers and downloading from users increased over time. The participants of an informal usability evaluation were particularly satisfied with the possibility to download complete albums, because the current peer-to-peer network clients only allow the download of single songs.

A typical problem of adaptive systems, like the music gathering application, is the creation of the initial user profile. This problem is often referred to as the "bootstrap problem". Our application is able to generate a good initial user profile by analysing the metadata of the user's existing music collection. A first test revealed that many MP3 users have a collection of at least 300 MP3 files, of which about half contain usable metadata. The initial user profile generated from these MP3 files turned out to be quite satisfying and many users commented that the profile matched their preferences. A more formal user test is in preparation.

CONCLUSION

We described a music gathering application with an emotional interface character for mobile Internet-enabled MP3 players. The storage capacity and jukebox functionality of mobile MP3 players are not a critical issues anymore. However, the acquisition of new content remains a problem. We proposed a proactive music gathering application that automatically obtains music from the Internet based on the user's profile. The application's architecture consists of several agent and non-agent software components.

The numerous aspects of the status of the proactive music gathering application, such as the number of available servers, speed of the download and the availability of chart information are too complex to be visualized given the small screen size of a mobile MP3 player. Therefore a comic character face based on eMuu [1] is used to provide a natural and instant feedback to the user. We have implemented the user interface by using four tabs that resembles the steps the user has to perform to gather music.

REFERENCES

1. Bartneck, C. (2002). *eMuu - an embodied emotional character for the ambient intelligent home.* Unpublished Ph.D. thesis, Eindhoven University of Technology, Eindhoven.
2. Bratman, M.E. , Isreal, D.J., Pollack, M.E. (1988). Plans and Resource-Bounded Practical Reasoning. *Computational Intelligence, 4*(4), 349-355.
3. Compaq. (2002). *iPaq*, from http://www.compaq.com/products/handhelds/pocketpc/
4. Dennett, D. C. (1981). Intentional Systems. In J. Haugeland (Ed.), *Mind Design* (pp. 220-242): Bradford Books.
5. FreeDB. (2002). *FreeDB*, from http://www.freedb.org
6. Nokia. (2002). *Nokia Communicator 9290*, from http://www.nokiausa.com/communicator
7. Ortony, A., Clore, G., & Collins, A. (1988). *The Cognitive Structure of Emotions*. Cambridge: Cambridge University Press.
8. Palm. (2002). *PalmPilot*, from http://www.palm.com/
9. Psion. (2002). *Psion Revo*, from http://www.psionusa.com/PersonalMobility/Revo/index.html
10. Wooldridge, M. (2002). *Multi-Agent System*. Hoboken, NJ: John Wiley & Sons.