Experiences from the Design of a Ubiquitous Computing System for the Blind

Vlad Coroama
ETH Zurich
Institute for Pervasive Computing
8092 Zurich, Switzerland
coroama@inf.ethz.ch

CHI 2006, April 22-27, 2006, Montréal, Québec, Canada.
ACM 1-59593-298-4/06/0004.

Abstract
This paper presents the user interface experiences we had while developing an assistive system for the blind and visually impaired based on ubiquitous computing technology, the Chatty Environment. After introducing the system, it describes several issues encountered during user interface design and the chosen solutions. It then shortly presents the results of a conducted user survey.

Keywords
Ubiquitous computing, visual impairment, blindness, interface issues, user survey.

ACM Classification Keywords
H.5.2 User Interfaces: Auditory feedback.
J.3 Life and medical sciences.

Introduction
The blind and visually impaired encounter many difficulties in their everyday life. Some of these problems, like their difficulty of finding the way through an unknown airport terminal or university building, seem to be easily understandable by sighted people. Others though are less obvious, and sighted researchers are likely to learn of these problems only when speaking to the blind. One example is the great trouble a blind person encounters in supermarkets when shopping on her own, since all packed food feels similar. Without external help – as several blind persons

told us in interviews – they will only go to the local supermarket and buy the few items in known locations. Such difficulties seem to have a common cause – the lack of information about their surroundings that the blind experience. It is not only the shape of the surroundings, it is also that much of the information we perceive is written – like the way to the escalators or restrooms in open buildings, or the ingredients list, price and best-before-date on any supermarket item.

Bearing the difficulties encountered by blind people in mind, we proposed the paradigm of a chatty environment, a ubiquitous computing system designed to help blind and visually impaired persons to lead a more independent life [3]. The main idea behind the system is to enhance the visual information existing around us by other means of information that can be experienced by the visually impaired. After informal discussions with a blind colleague (who accompanied us along the whole project), we derived a first set of requirements and developed a first version of the system. Then we conducted more formal interviews with nine potential users of the system, to which we have been pinpointed by the Swiss Association of the Blind.

The next section briefly presents the first set of requirements we started our work as well as the system and some of its technical aspects. The following sections present the design issues and the user survey results, respectively.

The Chatty Environment

The initial ideas were refined in a first set of discussions with potential users. The results of these interviews, presented in [3], can be summarized as follows. In terms of provided functionality, there seem to be two main requirements to the system. First, the user's perception of the surroundings should be enhanced by telling her which entities she is passing by. This seems to be the most important user requirement – to have an extension of their own world perception by having the environmental entities in their immediate neighborhood announced to them. Second, physical navigation to points-of-interest is a much desired aid. In terms of usability, this first round of discussions also led to some results, like the user's desire of not having to pinpoint to a certain location to get the desired information (this being, obviously, especially difficult for the completely blind interviewees).

In a nutshell, the Chatty Environment prototype consists of several components. Electronic markers are used to tag real-world objects. The user carries a mobile device that senses the markers placed in the environment. The markers store either information about the object they correspond to (embedded in an XML file) or at least a unique ID that can be resolved to a Web address containing that information. Some of the markers can even sense dynamic data from the physical environment of the object they are tagging, thus keeping a highly dynamic and up-to-date XML file about the object's state. The tags make real-world objects detectable by the user's mobile device. When the user comes into the vicinity of an enhanced object, his mobile device discovers it, receives the information about the object, and presents the object to the user. The user can then logically navigate through the object to gather more information about it, as described below.

For the moment, either RFID Tags (precisely, μ-Cips [4]) or Berkeley Motes [2] can be used to tag real-world objects. RFID tags only store a unique ID, that is resolved by the user's WiFi-enabled mobile device to a Web address. There, the static information about the tagged object is stored. In contrast, Berkeley Motes not only store the XML file themselves, but are also able to read sensory input and dynamically embed the data in the XML information file, as it becomes available. The software being modularly programmed, other tagging methods can be easily added. As mobile device, we use an HP iPaq 5450, with integrated WiFi connectivity. The device connects to motes placed in the environment and reads µ-Chips. The text embedded in XML information files is read to the user via a commercial text-to-speech engine installed on the device.

Using µ-Chips and Berkeley Motes as tagging devices in the Chatty Environment allows a large degree of flexibility and the testing of different scenarios that have been pointed out by potential users as being especially useful. The motes, with a transmission range that can be set between a few centimeters and up to 100 meters, are used to mark large or important objects that have to be detected from a distance, roughly matching the way a sighted person would discover these objects. The option of setting the transmission range when programming the motes proved to be especially useful. In many cases using the maximum transmission range seemed undesirable since it would have probably confused the user. A very short transmission range on the other hand (µ-Chips have a range of under 5cm) allows the implementation of a different kind of scenarios characterized by a high number of small items in close physical vicinity.

In our research lab, we have prototypically realized two such scenarios that will be presented below. They both reveal the user some part of his surroundings. Physically navigating the user to a destination, although an important requirement from the interviewed blind and visually impaired, was not implemented in our prototype. We decided to focus on the environment disclosure rather than the navigation part because it seemed from a user perspective to be at least as important as the navigational part and it was less explored in related projects (see section on related work).

User Interface Design Issues

While moving through the environment, information about objects in the user's neighborhood is gathered by her device. However, a series of questions arise regarding to when to present this information, when to delete it, how to present it to the user and how to let the user logically navigate through an object or among several objects.

Push/Pull Model

At the beginning, the Chatty Environment had only a *push model* for presenting new objects to the user, i.e., as soon as the user's device senses a new object in the neighborhood, the user is informed of its presence. This seems the logical way to go – inform the visually impaired users about their surroundings without requiring them to take explicit actions. This paradigm also maps the way sighted people gather information about their surroundings – unobtrusively from the corner of the eye, so to say. However, we soon realized that while this paradigm works well for large and relatively sparsely distributed objects, it fails for small objects with a high density like supermarket items. The user will certainly not want some hundreds or thousands of different products be announced to him as he enters the supermarket. Here, a *pull model* seems more appropriate – having the user explicitly choose an item, then supplying him the needed information. Technically, the two tagging systems work identical. As soon as a tag is discovered, the user is informed about its existence. However, we used the fact that the µ-Chips have a reading distance of only a few centimeters. We placed the reading antenna (that is connected to the user's mobile device in his backpack) on one of the backpack's straps near the user's right ear. Thus, from the user perspective, only when she brings such a product to her ear it starts talking to her, the user being the one explicitly initiating the communication.

Logically Navigating through Objects

After a new tag has been discovered, the information XML file is downloaded and the new object presented to the user. The information is organized as a tree that can be navigated by the user. The tree's root is the designation of the object. It can be freely chosen, but will usually be the object's textual description in a meaningful way for the user. Examples might be "Tram no. 9", "Ticket vending machine", or "Spice: Oregano".

When a newly discovered object is presented to the user, he will hear via earplug the text "New object" followed by the object's designation, e.g. "New object: ticket vending machine". (Texts are converted to speech by the commercial text-to-speech engine installed on the mobile device.) To understand how the user may logically navigate through an object or between several objects, let's take a look at the data model on the user's device (see Fig. 1). On the top level, there is a list of all objects active in the system. The information for an object is organized as a tree with the object's designation at its root. Trees can be of any height – the picture shows a tree of height two.

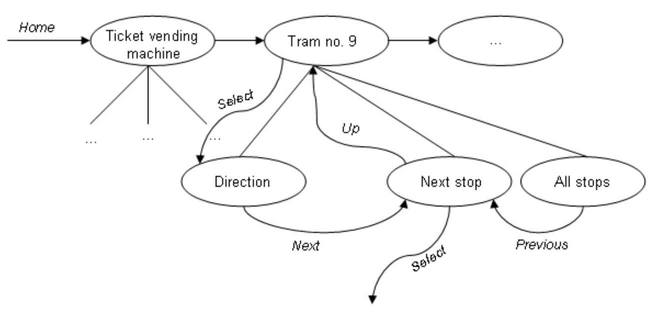

figure 1. Data model example.

Every time a new object is discovered, its designation is presented to the user. Then, the new object is added in front of the list of active objects. The user can always go down the branch of a tree if she presses the *select* button while the text of the father knot is being read or 2 seconds afterwards. For example, if pushing the button while the system reads "tram no. 9" or shortly afterwards, the user will get one level deeper into the tree and will hear the next level of information for that object: "direction… [2 seconds pause]… next stop… [2 seconds pause]… all stops". Here again, she may choose to go one level further down by pressing the *select* button. These knots already being leafs in this example, the user will hear the desired information. If the user misses the opportunity to press the button, he may always navigate back on the same level by using the *back* button. He also has the opportunity of speeding up the process by pressing the *next* button. Going up one level in the tree can be done any moment by the *up* button. Also at any moment, the user might press the *home* button to get to the first object in the list (which is the last discovered). After pushing the *home* button, all active objects in the system will be read, starting from the newest one, until the user chooses one of them to 'dive' into.

Deleting Objects from the System

This obviously raises the question of when to erase an object from the user's system. Since the mobile device is continuously polling the environment for markers, it always knows whether an object is still in range or not. While the object is still in range, no action will be taken; especially, the object's designation will not be presented to the user as being a new'one. A design decision that had to be taken, however, was how long to keep an object active after its signal has been lost by the mobile device. There are two reasons for not deleting objects at once. First, the user could move on the border of the object's signal radius, losing the signal repeatedly, but would not want the same object to be presented again and again as a new one. Second, even after not being in range any more, the user might want to get some information about an object. On the other hand, keeping objects active for too long might present the user an outdated world view and would contradict the paradigm of the Chatty Environment – presenting the user her neighborhoods. A trade-off had to be made here, which seemed rather difficult to be done globally. We learned that for motes a balanced default value would be 30 seconds. For RFID tags, the default is set to 5 seconds. Every single object might overwrite this default though (by an attribute that can be set in the XML information file), since enforcing a global value for every possible application doesn't seem a viable option.

User Survey

To test the Chatty Environment, potential users have been presented with two scenarios. In the first, users would move in an unknown environment (walk along corridors in a university building) and objects along the way would reveal themselves to them, such as the several offices and other rooms along the way, or objects like the coffee machine and the refrigerator in the kitchen. A total of 14 objects have been tagged by motes. The users had no specific task (like finding out where X is, or finding their way to Y), they could freely play with the system. The second scenario simulated a supermarket. The users were presented a shelf that contained five different sorts of (packed) spices and five different sorts of rice. These are items that feel the same and blind people are typically not able to distinguish them while shopping. For some products we had several packages with different best-before-dates. Users had to find a specific item and – if unhappy with a too close best-before-date – find another package that will be usable for a longer period of time. Nine potential users – 5 women and 4 men, in the range of 30 to 81 years – tested the system. Five of them are completely blind, four have an impairment level of between 60 and 98%. Only two interviewed persons own a Personal Digital Assistant for the blind, but most of them had held such a device in their hands before.

After testing the system, interviews were conducted in two steps: All interviewed first answered a questionnaire comprising 20 questions, ranging from general information about their age, profession or impairment grade to precise questions about how they experienced the system. The interview was also based on the open-end principle, each participant being able to add any information or suggestion considered to be relevant. The interviews were about one hour long.

The environment endlessly speaking to the user, telling her about the surroundings may seem annoying to most sighted people. However, most interviewed were not disturbed by objects that "speak". They rather experience a substantial lack of information today and welcome any new information, especially the completely blind users. As one blind participant put it, "there can never be too much information". Two said that too much information could ultimately become annoying and make them feel uncomfortable. Thus, they would prefer to be able to adjust the number of objects that speak, i.e., the information density, depending on their actual needs.

Speech was the preferred output medium for almost all interviewed people. Some of them could find utility in additional signaling techniques, such as vibration or non-speech audio signals (i.e., beep). However, the normal hearing of the user must not be altered by the system. Blind need stereometric hearing to orient themselves, e.g., in order to determine the direction of moving obstacles. Therefore, any kind of headphones or earphones used have to let environmental sounds muddle through. All participants liked to carry the device in the backpack, in order to keep the hands free for other functions.

The logical navigation through objects was quickly understood by all testers. The tree structure seemed natural. However, the 5-way navigation joystick was too cumbersome for some and they wished a more robust way to push the *select* button, without the . Thus, we added a dedicated button for the A challenge for future versions will be to reduce the navigational freedom, while keeping the functionality.

Related and Future Work

In recent years, several research projects using GPS-based guidance for the blind have emerged, such as the "Personal Guidance System" [7] or Drishti [5]. One of the few systems aiming similar objectives and applying a similar approach as the Chatty Environment is the "Navigational Assistance for the Visually Impaired" (NAVI) [6]. The user's portable device combines a CD player with a mobile RFID tag reader, reading the tags spread in the environment. Every tag triggers the corresponding track on the CD to be read to the user. However, NAVI seems not to scale well due to the static data stored on the CD that quickly becomes outdated. [8] also proposes the use of RFID tags to help the blind orient themselves, but is rather focused on navigation than environment disclosure. Both projects are limited to RFID tags and their short transmission range. A newer and more flexible system, conceived for helping blind in public transportation scenarios, is Ubibus [1].

As of our system, we received valuable ideas from the users regarding the navigational part of the Chatty Environment – i.e., on how routes should be described, which should be preferred over others, what obstacles should be avoided, etc. These results, outside the scope of this paper, will be used for the development of the navigational part of the Chatty Environment.

References

[1] Banâtre, M., Couderc, P., Pauty, J., and Becus, M. Ubibus: Ubiquitous Computing to Help Blind People in Public Transport. In: *Proc. MobileHCI 2004*.

[2] Berkeley Motes. www.xbow.com/Products/Wireless_Sensor_Networks.htm.

[3] Coroama, V., Kapic, T. and Röthenbacher, F. Improving the Reality Perception of Visually Impaired through Pervasive Computing. *Advances in Pervasive Computing*, OCG Press (2004).

[4] Hitachi µ-Chip. www.hitachi.co.jp/Prod/mu-chip/.

[5] Helal, A., Moore, S., and Ramachandran, B. Drishti. An integrated navigation system for visually impaired and disabled. *Proc. 5th Symposium on Wearable Computer* (2001).

[6] Patch, K. Radio tags give guidance. www.trnmag.com/Stories/2003/092403/Radio_tags_give_guidance_092403.html

[7] PGS webpage. www.geog.ucsb.edu/pgs/main.htm.

[8] Willis, S., Helal, S. RFID Information Grid for Blind Navigation and Wayfinding. In: *Proc. of the 9th IEEE Symposium on Wearable Computers* (2005).