Using Context-awareness to Support Adaptive Multimodal Mobile Notification

JANET L. WESSON[1]
Nelson Mandela Metropolitan University, Port Elizabeth, South Africa

WILLIAM BRANDER
Bosasa Research and Development Laboratory, Johannesburg, South Africa

Abstract

Multimodal mobile notification can enable natural and implicit notification in a mobile context by adapting the output modality according to the user's context. There are several existing models for adaptive mobile notification using context awareness. A comparison of two of these models identified several shortcomings. The IBM INS model was selected as the most appropriate model and extended to support context history and mode switching. This paper discusses the proposed model for adaptive multimodal mobile notification together with the development of a prototype, called Mercury, based on this model. The aim of this research was to determine the usefulness and accuracy of adaptive multimodal mobile notification using the user's context to adapt the mode of notification. Mercury was evaluated using an extensive field study conducted over a period of two months. The results obtained show that the users found the adaptive mobile notification service to be useful and accurate, but that they had some concerns about the level of interruptions.

Categories and Subject Descriptors: H5.2 [Information Interfaces and Presentation]: User Interfaces - Evaluation/methodology; Graphical user interfaces; Interaction styles; Screen design; User-centered design
General Terms: Human factors, Experimentation, Measurement, Verification, Design
Key Words: adaptive systems, context-awareness, mobile notification services, multimodality, usability evaluation

IJCIR Reference Format:

Janet L. Wesson and William Brander, Using Context-awareness to Support Adaptive Multimodal Mobile Notification. International Journal of Computing and ICT Research, Special Issue Vol. 4, No. 1, pp. 11 - 19. http://www.ijcir.org/Special-Issuevolume4-number1/article2.pdf.

1. INTRODUCTION

Computer mediated communication such as email has received a lot of attention in recent years. Email allows messages to be sent and received almost instantly, regardless of the location or availability of the receiver. The convenience and benefits of email have made people reliant on this method of communication. A system whereby email notification can be automatically sent to a user's mobile phone can potentially have several benefits. If the user is unable to read a text message for some reason, e.g. because he or she is driving a car, multimodality could be considered. It has been shown that multimodal interaction is more effective than single mode interaction in certain contexts, such as a pedestrian environment [Jöst et al. 2005].

[1] Author's Address: Janet L. Wesson, Nelson Mandela Metropolitan University, Port Elizabeth, South Africa; William Brander, Bosasa Research and Development Laboratory, Johannesburg, South Africa

International Journal of Computing and ICT Research, ISSN 1818-1139 (Print), ISSN 1996-1065 (Online), Special Issue, Vol.4, No.1, pp. 11 - 19, October 2010.

This paper discusses the development of a model for adaptive multimodal mobile notification. This model was based on the IBM INS model [Bazinette et al. 2001], which was extended to include three separate modules, namely an Input Module, an Output Module and a Context Module. This model was used to develop a prototype, called Mercury, to provide notification services to mobile users in either text or voice using context-awareness. The aim of this research was to determine the accuracy and usefulness of an adaptive multimodal mobile notification service. This research also aimed to identify any concerns that the users experienced, specifically relating to user control, privacy and interruptions.

The structure of this paper is as follows: Section 2 describes the characteristics of multimodal mobile notification and identifies the requirements for adaptive multimodal mobile notification. These requirements are used in Section 3 to compare two existing models for mobile notification and identify the benefits and shortcomings of each model. Section 4 proposes a new model for adaptive multimodal mobile notification. Section 5 discusses the development of Mercury, based on the proposed model. An extensive field study of Mercury was conducted, the results of which are discussed in Section 6.

2. MULTIMODAL MOBILE NOTIFICATION

Mobile notification can be defined as sending a text or multimedia message to a group of wireless subscribers. Notification is a form of push technology where information is transferred as a result of occurrence of an event. A wireless subscriber is a consumer that subscribes to events in which the user is interested. Once these events occur, the event is published to all those consumers who have subscribed to that event. An event is published by notifying all subscribed consumers using the most appropriate device, medium and mode of communication. The mobile phone has become the most popular mobile device in use today because of its convenient and personal nature. Since mobile phones are primarily communication tools, they can provide ideal support for notification using different modalities (e.g. text, voice or multimedia) [Barón and Green 2006].

There are several factors contributing towards the need for multimodal mobile services. Road safety has become a major concern since the advent of mobile phones [Brumby et al. 2007]. Driving and using a mobile phone without a hands-free kit is illegal in many countries. Multimodality can provide a convenient means for drivers to hear messages without having to read the text [Vilimek et al. 2007]. The number of mobile professionals is growing rapidly and these users are ideal candidates for multimodal mobile services [Colby 2002]. Multimodal systems have also been shown to provide higher levels of user satisfaction [Oviatt 2000]. In some studies it has been shown that having the option of switching modalities was enough to provide greater user satisfaction even if the user never actually switched modes [Brewster 2002].

2.1 Context-Awareness

Dey et al. [2001] define context as "any information that can be used to characterize the situation of an entity, where an entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves." The concept of using information collected by passive sources to alter the meaning of the data is referred to as context-awareness. A context-aware application takes the user's environment into account and uses that information to alter the way the data is presented.

Context can be acquired from several sources including the mobile device, the mobile device's network or a combination of both. For multimodal context-aware mobile applications, the application needs access to sensors that may be available on the device or from the network. If the sensors that provide context information are located on multiple devices, there needs to be a way to synchronize this information.

2.2 Requirements

Mobile applications need to support mobility and provide for a large number of subscribers and publishers. Notifying users of content applicable to their current location is significantly more effective than notifying them of content which is not applicable to their current environment [Steinfield 2003]. The context of use can also determine which mode of communication the user would prefer to use [Love and Perry 2004]. In a noisy environment, if the system can determine that the ambient noise level is too high for voice output, it can automatically switch to a textual mode of communication.
A set of criteria for an adaptive mobile notification service were derived from literature, as follows [Brander and Wesson 2006]:

1. Abstraction (context acquisition must be abstracted from context use).
2. Flexibility (new sensor technology must be supported without re-designing the model).
3. History (contextual history must be supported).
4. Distributed Communication (distributed communication between sensors, data stores and modules must be supported).
5. Synchronization (multiple sensors must be able to synchronize their times in order to provide consistency).
6. Mode switching (the model must allow for dynamic switching of input and output modes).
7. Mode independence (the model must not rely on any particular mode in order to operate).

In order for a model to be extensible, it also has to support scalability, namely the ability to add new notification sources using a plug-in approach. This allows for new services to be added without changing the design of the model. The next section discusses two existing models which support adaptation and extensibility.

3. EXISTING MODELS

Several models for mobile notification using multimodality and context awareness were reviewed. Based on the above criteria, two of these models were selected as candidate models. These models were the W3C Multimodal Interaction Framework and the IBM Intelligent Notification System (IBM INS), which are discussed and compared in the next two sections.

3.1 W3C Multimodal Interaction Framework

The W3C Multimodal Interaction Framework [Larson and Raman 2003] identifies the major components that a multimodal system should possess and describes how these components relate to each other (Figure 1).

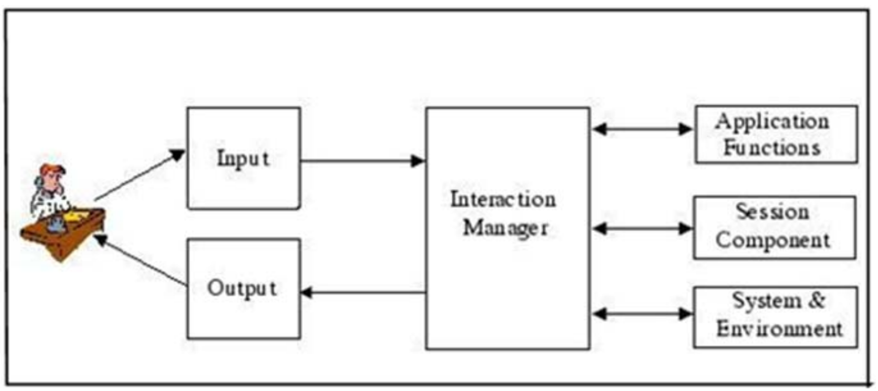

Figure 1 W3C Framework for Multimodal Notification

The input is provided by multiple input modes and devices. The combined input from all devices and modes is then passed to the Interaction Manager. The Interaction Manager takes the input and combines it with the application's functions, any session information and information about the environment of the system and the user. The model does not, however, cater for more than one service being implemented at once as multiple application components would be needed.

The System & Environment Component allow multiple sensors to be used in determining the user's context. The system and environment information is responsible for adapting the output mode based on the user's context. After the Interaction Manager has finished computing an output based on the input, it passes the output to the Output Component.

The Output Component accepts an output string as input, which is then processed to determine which output mode will be used for the notification. The Output Component is responsible for presenting the data to each user, which could be in SMS format, or via a web page or using voice.

3.2 IBM Intelligent Notification Service

The IBM Intelligent Notification System (IBM INS) [Bazinette et al. 2001] is an adaptive mobile notification service model which allows users to specify events which trigger notifications (Figure 2). These events may require the system to aggregate data from various content sources. Each content source has its own Content Adapter, which checks for new content and then formats the new content in a form suitable for use by the service.

The Content Adapter then informs the Trigger Management Service of the new content, which checks if any users are interested in that event. In this case, the Trigger Management Service requests a notification from the Universal Notification Dispatcher. The Universal Notification Dispatcher then requests the user's context from the Secure Context Service and makes the notification in the user's preferred output mode.

The IBM INS model does not support mode switching as user input is limited to one mode per operation. User registration is done solely in one mode (from a web interface) and the model is thus dependent on a particular mode.

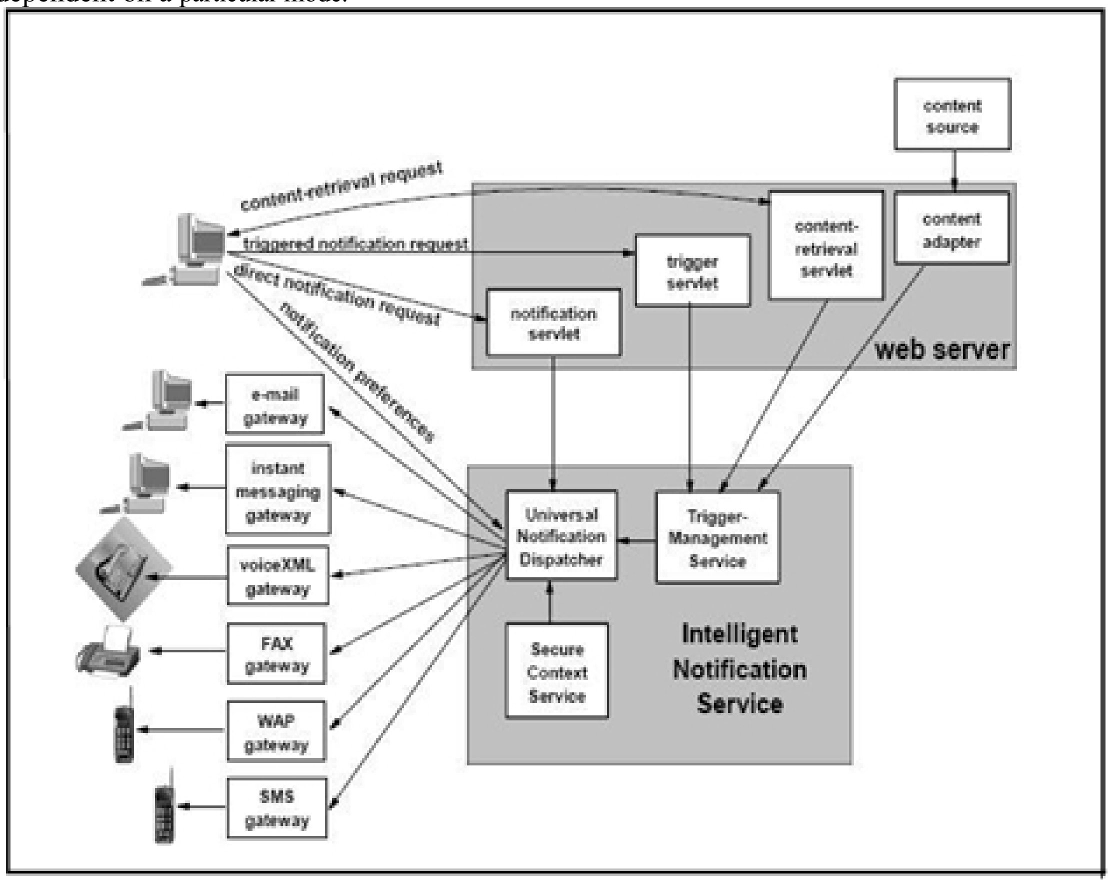

Figure 2 IBM Intelligent Notification Service

4. PROPOSED MODEL

The IBM INS model supports most of the requirements for an adaptive mobile notification service except for context history, time synchronization and mode functionality. The proposed model is an extension of the IBM INS model and comprises four core modules and a set of Content Sources [Brander and Wesson 2006]. The Content Sources are responsible for determining when new content arrives and parsing that content into a suitable form for further use. The four modules are the Trigger Management Service, the Input Module, the Context Module and the Output Module (Figure 3).

The Input Module is responsible for taking input from the user and converting it into a standard format, which is device and mode independent. The Interaction Manager is responsible for combining all the input from the various devices and modes into one operation. Once the Interaction Manager has combined all the inputs, it passes the result to the Trigger Management Service.

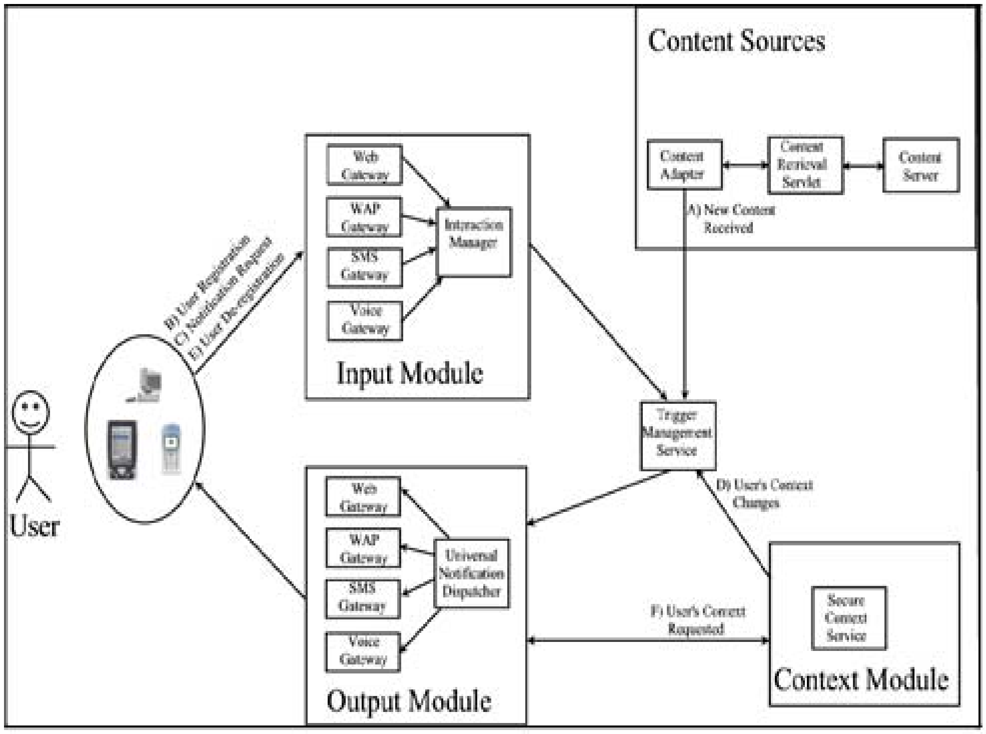

Figure 3 Proposed Model for Adaptive Multimodal Mobile Notification

The Content Sources are the various sources of content that are notified to the users, such as mail servers, news web sites or stock market sites. Once new content is available, the Content Retrieval Servlet passes the content to the relevant Content Adapter. Once the Content Adapter has converted the content into its own format, it passes the new content to the Trigger Management Service.

The Secure Context Service is responsible for determining each user's context and passing that information to the Output Module. The Secure Context Service consists of various sensors, history stores and a Context Request Mediator. The Context Request Mediator determines which sensors are applicable to the user and combines all the separate sensor information into a single, high level context.

The Trigger Management Service is responsible for initiating the notifications based on events and consists of a Trigger Manager, an Event Matching Engine and a set of Data Stores. Each trigger is associated with a particular event or content type such as new email arriving on the mail server or a web site being updated.

The Output Module consists of a Notification Controller, a Preference Engine and a Notification Request Queue. The Preference Engine contains each user's notification preferences and the notification preferences for each service. This allows notifications to be filtered based on content type and occurrence.

When the Notification Controller receives a notification request from the Trigger Management Service, it requests the user's context from the Secure Context Service and the user's preferences from the Preference Engine. It then places the notification on the Notification Request Queue together with the preferred output mode. The Notification Dispatcher retrieves the message from the queue and selects the appropriate gateway(s) to dispatch the notification.

5. MERCURY PROTOTYPE

A prototype, called Mercury (after the Greek winged messenger of the gods), was implemented as a proof-of-concept of the proposed model. Mercury was designed to provide email notification as well as notification of network state changes. User preferences were stored as a link between the mode of notification and the user's context, as discussed in Section 5.3 below.

5.1 Obtaining Sensor Information

Several sensors are used in order to determine a user's context. Certain sensors are more applicable than others, for example calendar information could be more relevant to a user's context than the air-pressure around the user. In order to cater for this, a weighting value is assigned to each sensor with a higher weighting, implying more relevance.

If a particular sensor is not available, the weighting of each of the remaining sensors is increased by the ratio of that sensor's weighting to the remaining sensors. The value of a sensor can be binary, boolean or numeric. Calendar information is used to determine whether or not the user is in a meeting (boolean). The value of a numeric sensor can range from 0 to 1, with a higher value being regarded as more significant than a lower value.

5.2 Determining Context

The Secure Context Service uses multiple sensors to determine the users' context. Each sensor has a weighting (W_i) assigned to it, which is used to determine which specific context the user is in. If a sensor supports that the user is in a specific context, the weighting of that sensor (W_i) is multiplied by the value of the sensor (V_i) to determine the weighted sensor value (W_iV_i). A total confidence (C_i) is calculated for each possible context and the context with the highest confidence is determined to be the most likely context, as follows:

$$Ci = \sum WiVi$$

The set of six possible contexts was determined as follows [Abowd et al. 1999]: At desk, In a meeting, Sleeping, Home (Other), Work (Other), and Driving. Each user's contextual history is stored to facilitate time-based inferences based on the user's context. This historical information can be used to determine additional information from existing contextual information. For example, the user's speed can be determined from the history of the user's location. The user's context was determined by using a combination of the following information: GPS location, ambient light level, ambient noise level and the user's calendar.

When a user's context is requested, a SMS is sent to trigger the Context Acquisition Midlet running on the user's mobile phone. The Midlet then does the following: Activates the phone's camera and takes a picture; queries the GPS for the user's current location and movement; records a two second sound clip; and accesses the user's calendar to get the current appointment information.

The Midlet then sends the information to Mercury which processes this information to determine the user's context. The picture is processed to determine whether the user is under fluorescent light by determining the whiteness of the lighting in the picture. The sound clip is analyzed to determine the ambient noise level, classified as being either loud or soft. The calendar information is used to determine if the user is scheduled for a meeting and the GPS information is used to determine if the user is in a car or not.

Once the sensor information has been determined, the overall context of the user is calculated. For example, consider the following sensor information: The user's calendar says the user is in a meeting (value=1); the ambient lighting says he is in a office environment (value=0.8); the ambient noise is high (value=0.2); and the person is stationary (value=1).

Using the weightings in Table 1, the confidence of each context is determined as follows: Confidence (In a meeting)=0.9*1+0.1*0.8+0.15*0.2+0.15*1= 1.16; and Confidence (Other)= 0.5*0+0.5*0.8+0.5*0.2+0.9*0= 0.5. Confidence (In a meeting) is higher than Confidence (Other), so the user's context is determined to be "In a meeting".

Table 1 Example Sensors, Contexts and Weightings

Sensor	In a Meeting	Other
Calendar	90%	50%
Light	10%	50%
Sound	15%	50%
GPS	15%	90%

5.3 Determining Notification Mode

User preferences are stored as a link between the user's context and the mode of notification. Previous research has shown that most users are incorrect in their initial choice of output mode [Chen and Kotz 2000]. A single-layered neural network was therefore used to keep the user's preferences up to date. The neural network was based on a typical pattern recognition neural network.

The neural network uses the set of initially chosen preferences as the base training case and training continues for a period of two weeks. During this training period, every time a notification is made, the user is asked if the notification was made in the correct mode. If not, the user is asked what the most appropriate mode should be. The answers to these questions are fed back into the neural network, which then adjusts the weightings for each mode. This has the effect of changing the user's mode preferences with actual usage data.

6. EVALUATION

The goal of the evaluation was to determine the accuracy and usefulness of adaptive multimodal mobile notification in a real-world environment. A field study was therefore selected as the most appropriate evaluation methodology to obtain this information [Garzonis 2005].

6.1 Evaluation Metrics

In order to achieve the goals of the evaluation, several metrics were identified:

- How often was the correct context determined?
- How often was the correct mode of notification used?
- Did the users regard the notifications as useful?
- Did the users feel comfortable with the level of control provided?
- Did the users have any concerns about being interrupted?
- Did the users have any privacy concerns?

6.2 Research Method

Usability evaluation for desktop environments is not well suited to mobile environments due to the lack of situational influences [Duh et al. 2006]. Garzonis [2005] showed that field testing is more effective for identifying issues related to context of use than traditional laboratory testing. It was therefore decided that a field study would be conducted over a period of two months with the users being required to complete several user satisfaction questionnaires during the period.

6.3 Participant Selection

A sample population which approximated the target user population was selected for the field study. The sample population comprised young executives employed at the Bosasa Research and Development Laboratory. There were a total of 74 participants, all of whom were between the ages of 23 and 31. Fifty eight percent (43) of the participants were female and 42% (31) were male. Each of the participants in the field study was provided with a mobile phone, which allowed Mercury to obtain the necessary sensor information whenever a request for a user's context was made.

An interview was conducted with each participant every 3 weeks during which he/she was also asked to complete a user satisfaction questionnaire. Upon user registration, each participant was asked in which mode of communication he/she would like to receive notifications. For a period of two weeks after the initial registration, the participants were asked whether or not the notifications were made in the correct mode. Each response was used to further train the neural network. Once the training month was completed, the following month was used to determine the accuracy and usefulness of the adaptive mobile notification service.

6.4 Results

The initial user preferences revealed that 84% (62) of participants wanted all of their notifications to be made using text except for when they were driving. Almost all participants wanted their notifications to be voice-based whilst they were driving. Seven percent (5) of the participants wanted all of their notifications to be made in text regardless of the context of use.

A total of 8,573 notifications were performed by Mercury over the two month evaluation period. Of these notifications, 79% (6,773) were made using the correct context (Table 2). We can therefore conclude that Mercury was highly accurate in determining the correct context.

Table 2 Accuracy of Determining Confidence

	Text	Voice	Total
Correct context determined	66%	13%	79%
Incorrect context determined	10%	11%	21%
Total	76%	24%	100%

A comparison of the initial preferences chosen by the participants with the preferences learned by the neural network was made to determine how accurate the initial preferences were. The results showed that in all contexts (except driving), the learned preferences were different from the initial user preferences (Table 3). The percentage of participants who changed their initial preferences was found to be above 50% for all contexts except driving.

Table 3 Accuracy of Initial and Learned Preferences

Context	Unchanged	Changed
At desk	43%	57 %
In a meeting	11%	89 %
Sleeping	20%	80 %
Home (Other)	30%	70 %
Work (Other)	43%	57 %
Driving	73%	27%

Research has shown that users have several concerns with respect to context-aware applications, including user control, frequency of interruptions, security and privacy [Barkuus and Dey 2003]. The participants were presented with a user satisfaction questionnaire every three weeks in order to measure the effect of these concerns. Each question had a 10-point Likert scale, which asked the participants to rank their concerns from None (1) to Highly concerned (10). The results revealed that, as the field study progressed, the control and privacy concerns become less pronounced while concerns about the number of interruptions increased (Figure 4). concern could possibly be addressed by developing filtering mechanisms to reduce the number of mobile notifications received by users.

7. DISCUSSION

The context of a mobile phone user is an important factor in adaptive multimodal mobile notification. The user's context can be obtained by combining information from several sensors and data sources that are available on a typical mobile phone, such as ambient light level, noise level and GPS location. This context can be used to determine the most appropriate output mode to use for mobile notification. Initial notification preferences can be obtained from the users, but these are not very accurate, with the exception of driving, where voice is preferred. An adaptive multimodal mobile notification system will need to learn the users' actual preferences, as these are more accurate than their initial preferences. User control and privacy are not serious concerns, but the level of interruptions is a key factor which needs to be taken into consideration when designing mobile notification systems.

8. CONCLUSIONS

Mobile notification services can provide several benefits, but need to support context-awareness. This paper has proposed a model that meets the requirements for adaptive mobile notification using context-awareness and multimodality. A prototype, called Mercury, was developed as a proof-of-concept of the proposed model. Mercury was evaluated using a field study conducted in a live environment for a period of two months.

The results of the field study clearly showed that the participants regarded Mercury as highly accurate and useful. The results also showed that the users' initial preferences were often wrong and that the learned preferences were more accurate. More research is needed, however, to develop filtering mechanisms to address the users' concerns about the number of interruptions.

9. REFERENCES

ABOWD, G. C., K., D. A., BROWN, P. J., DAVIES, N., SMITH, M. & STEGGLES, P. 1999. Towards a Better Understanding of Context and Context -awareness. Handheld and Ubiquitous Computing, 304-307.

BARKUUS, L. & DEY, A. 2003. Location-Based Services for Mobile Telephony: a Study of Users' Privacy Concerns. INTERACT 2003, 9TH IFIP TC13 International Conference on Human-

Computer Interaction. Zurich, Switzerland, ACM Press.

BARÓN, A. & GREEN, P. 2006. Safety and Usability of Speech Interfaces for In-Vehicle Tasks while Driving: A Brief Literature Review. University of Michigan Transportation Research Institute.

BAZINETTE, V., COHEN, N. H., EBLING, M. R., HUNT, G. D. H., LEI, H., PURAKAYASTHA, A., STEWART, G., WONG, L. & YEH, D. L. 2001. An Intelligent Notification System. IBM Research Report RC 22089.

BRANDER, W. & WESSON, J. L. 2006. Implementing Adaptive Mobile Notification Services: A Model Based Approach. Southern African Networks and Applications Conference (SATNAC 2006). Spier, Western Cape.

BREWSTER, S. 2002. Overcoming the Lack of Screen Space on Mobile Computers. Personal and Ubiquitous Computing, 6, 188-205.

BRUMBY, D. P., SALVUCCI, D. D. & HOWES, A. 2007. An Empirical Investigation into Dual-Task Tradeoffs while Driving and Dialing. IN RAMDUNY -ELLIS, D. & RACHOVIDES, D. (Eds.) 21st BCS HCI Group Conference (HCI 2007). Lancaster University, UK, British Computer Society.

CHEN, G. & KOTZ, D. 2000. A Survey of Context-Aware Mobile Computing Research. Technical Report: TR2000-381.

COLBY, J. 2002. Multimodality: The Next Wave of Mobile Interaction. Speech Technology, 7, 32.

DEY, A. K., SALBER, D. & ABOWD, G. D. 2001. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-aware Applications. Human Computer Interaction, 16.

DUH, H. B. L., TAN, G. C. B. & CHEN, V. H. H. 2006. Usability Evaluation for Mobile Devices - A Comparison of Laboratory and Field Tests. MobileHCI '06. Helsinki, Finland.

GARZONIS, S. 2005. Usability Evaluation of Context-aware Mobile Systems: A Review. 3rd UK-UbiNet Workshop.

JÖST, M., HÄUßLER, J., MERDES, M. & MALAKA, R. 2005. Multimodal Interaction for Pedestrians: An Evaluation Study. Proceedings on the 10th International Conference on Intelligent User Interfaces. San Diego, California.

LARSON, J. A. & RAMAN, T. V. 2003. W3C Multimodal Interaction Framework.

LOVE, S. & PERRY, M. 2004. Dealing with Mobile Conversations in Public Places Some Implications for the Design of Socially Intrusive Technologies. ACM CHI 2004. Vienna, Austria.

OVIATT, S. 2000. Multimodal System Processing in Mobile Environments. 13th Annual ACM Symposium on User Interface Software Technology.

STEINFIELD, C. 2003. The Development of Location Based Services in Mobile Commerce.

VILIMEK, R., HEMPEL, T. & OTTO, B. 2007. Multimodal Interfaces for In -Vehicle Applications. J. A. Jacko (Ed.), Human-Computer Interaction, Part III, HCII 2007, LNCS, 4552, 216-224.