WAP - Designing for Small User Interfaces

Albrecht Schmidt, Henning Schröder
TecO, University of Karlsruhe
76131 Karlsruhe, Germany
{Albrecht, schroder}@teco.edu

Oliver Frick
SAP Corporate Research, CEC Karlsruhe,
76131 Karlsruhe, Germany
o.frick@sap.com

Keywords
WAP, WML, mobile personal devices, small UI design

INTRODUCTION

Current and upcoming WAP-capable mobile phones introduce new user interfaces where standard methods for application design often fail. Automated translation of HTML to WML produces screen layouts and input mechanisms that are often not useable on a mobile phone.

In this paper we suggest a structured approach for building WAP application and in particular the development of the user interface. First we give a brief summary on the discriminating feature of a phone interface and on mechanisms to build applications on WAP devices. Then we provide guidelines on how to port from Web to WAP and how to develop WAP-Applications. We finally introduce the WAP application that led to our findings.

WAP-DEVICES

An increasing a number of WAP devices become available today. In contrast to desktop machines these devices do not form a homogeneous platform in terms of UI and usage. This paragraph gives a brief overview on our findings concerning different UIs and the support for WAP/WML.

User Interfaces

WAP devices have a variety of interfaces; output can be displayed as text, formatted text, or graphics. Input is provided by touch screens and/or buttons.

Displays are in general much smaller than that on PCs, however the size varies very much. The Ericsson MC218 for example has a 640 x 240 pixel display whereas the Nokia 7110 has only 96 x 65 pixel. This is a ratio of about 24:1 whereas workstation screens differ only by a ratio of 6:1 (1600x1200 compared to 640x480). This large differences in screen space makes it almost impossible to design an application so that it fits all WAP-devices [1].

The input mechanisms provided by WAP-devices are not coherent either. PDA like devices offer point and click on a touch screen whereas most phones rely on a set of buttons. Link navigation with a button interface however is more difficult than point and click. Scrolling up and down is supported very well on most devices (e.g. scroll wheel on Nokia 7110) whereas often the display width is fixed and no left right scrolling is provided. Text input is difficult and time consuming on most phone-sized devices, whereas inputting numbers is supported very well and users are accustomed to that. PDA like devices on the other hand typically come with alpha-numeric keyboards.

Applications, WAP, WML and WMLS

PDA like devices as well as phones offer a number of applications beyond phoning, such as messaging, calculator, address book, and games. For these applications, and to some extent even for phoning, the usage of devices from different manufactures (or even for different models of one manufacturer) does not follow the same rules. With the introduction of the WAP standard a manufacturer independent platform for the development and distribution of applications for WAP-phones and other WAP-devices is given. Applications are written in WML and or in WMLS and constitute of one or more decks consisting of on or more cards. The resource is provided on a server and can be downloaded by the user.

In our experiments we observed that most devices support only a subset of the specified markup tags and that the ways in which these tags are interpreted are also different. This makes it impossible to foresee the appearance of an application on the users screen; the difference observed is significantly larger than the differences observed with applications in different standard Web browsers on desktop machines.

DESIGNING AND DEVELOPING WAP-APPLICATIONS

From our experience in developing WAP-applications (a selected application is described in the next section) we derived a set of guidelines for the development of WAP-applications. In the next paragraphs these guidelines are stated as an informal procedure.

General Issues

The first and most important matter is to *identify the benefits for bringing an application to the WAP platform*. Supporting these benefits must be the prime goal of the application development. A potential benefit is, e.g., universal access to information that is vital while on the way (e.g. contact address, travel management).

The navigation space and implied interaction with the application should be reflected in the structure of decks and cards in the implementation. Wherever possible (if cards do not rely on user input) provide a maximal navigation space within one deck to avoid reloading during navigation.

If information is already provided on a Web site and should also be available on WAP the following approach can be used for porting: define a typical path or more general a graph and build a navigational structure for this path, which is then represented by a deck consisting of multiple cards. Used the titles and headlines of Web pages along the path

to have the content framework. Reduce text to a minimum and design information chunks that fit on the smallest devices you are designing for.

Input Design

Most current phones are optimized inputting numbers rather than text. Also direct point and click, as known from PDAs and PCs, is typically not given. Based on this the following recommendations for input design should be considered:

- use *numbers for input* when ever possible
- use *common abbreviations* like country codes
- if letters are used keep the input mechanisms in mind, e.g. *prefer first letter on key*
- *offer choices* (e.g. numbers, listboxes, radio-buttons, link-lists) or *default values* when applicable; even longer link list prove useable because of scrolling
- *provide labels to hardware* buttons were possible
- *use standard conventions* on buttons (e.g. back)

One basic issue is to reduce the need for user input as much as possible. Here customization of applications for the user provides one way. Also the cooperation between desktop interfaces and WAP-devices can be beneficial (cf. our case study described in the next section).

Output Design

Displays on mobile phones are in general rather small and provide only a low resolution and color depth. Thus, reading of larger texts is rather difficult whereas the display of small information chunks is much better. The usage of graphics does often not provide additional value because of the low resolution and poor rendering. The following puts attention to the most important issues:

- assess the *screen size and quality* of the target devices with text and graphics
- *reduce the output* by customizing to the users need [2]
- design *information chunks that are seen at once* on the screen, larger text blocks (more than 20 words!) within on card should be structured
- *horizontal scrolling is easy* on most devices, vertical is not!
- use *multiple cards in one deck* instead of very large cards or multiple decks

Considering the range of devices and their capabilities, it is advisable to design input and output for certain devices classes individually.

PERSONAL WAP - A CASE STUDY

Within the project *Personal WAP* we investigate the reduction of user interaction in WAP applications by providing users with a personalized WAP portal.

During our WAP usability studies we found inputting text, e.g. setting bookmarks on a phone, very inconvenient – but nevertheless we found it important to have a hotlist of links easy accessible and maintainable. In addition, we wanted to have mechanism to filter information both, personal and situation specific. Finally, we wanted to have access to short personal notes (e.g. an address, a shopping list, etc.) when leaving the office or home (a "WAP PostIt").

The solution consists of two interfaces - one for a standard Web browser and one for a WAP phone. The Web browser interface is used to define the personal WAP portal with links and notes. The WAP interface is a customized, user specific set of WML cards carrying the information previously selected or edited within the Web interface.

The Web Interface

A Web service where the user is identified by his phone number, access is granted by using a password. On this page the user can input links, news items, and notes that should be available on his personal WAP portal (Figure 1).

The WAP Interface

The WML deck which represents the personal WAP portal is generated from the information edited on the Web interface. The WAP URL to the personal WAP portal is fixed, whereas the content of the WML deck is dynamically generated. As the management of the WAP portal takes place on the Web interface, no tedious bookmark management on the phone is needed (Figure 1).

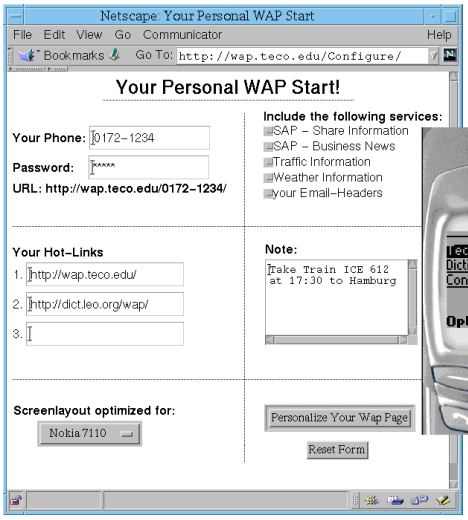

Figure 1

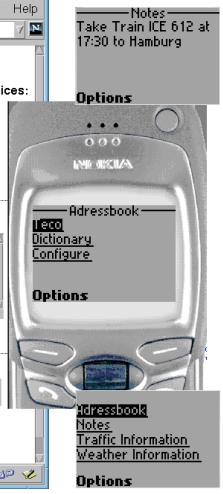

Figure 2

CONCLUSION

WAP applications require a new technique for user interface design. Prime design guideline is minimal user interaction during input and output. Suggested techniques for output reduction are WAP specific presentation as well as user driven customization. Suggested techniques for input reduction is service preselection, provision of choices, input by navigation, and focus on numeric input.

REFERENCES

1. Brewster, S., Cryer, P. Maximising Screen-Space on Mobile Computing Devices, in Proc. CHI'99, 1999.
2. Barret, R., Maglio, P., Kellem, C. How To Personalize the Web, in Proc. CHI'97, 1997.