

Enable User Input in Ubiquitous Computing

Dan Hong, Vincent Y. Shen
Department of Computer Science
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
{csdhong, shen}@cs.ust.hk

ABSTRACT

It is common for users to enter information during Web browsing. However, entering information using mobile devices is often difficult due to the limitations of input facilities on mobile devices. In this paper, we illustrate how a well-designed ubiquitous computing environment could improve the user input experience with the help of context from other sources. Based on a case study, context-assisted form filling, we demonstrate the feasibility of letting the computing environment contribute to a more positive mobile Web experience.

INTRODUCTION

Ubiquitous computing has become a hot topic since it was proposed in 1990's. In the ubiquitous environment certain related information, which is called "context", can be obtained from physically embedded or mobile sensors, deduced from other relevant context, and finally used in applications which can take advantage of this context.

Context is the relevant information that can affect a user's interaction behavior. In the mobile Web, context may include device capabilities, environment lighting level, network bandwidth, input styles, user preferences, interaction history, etc. How the context may affect a user's reaction is an interesting issue to study.

There is evidence that form filling is a popular Web activity. Many users fill out forms to check weather reports, stock prices, movie schedules, etc. The information requested is normally entered manually. However, form filling is quite difficult using a mobile device since instead of the keyboard and mouse, the device may support only a stylus and a keypad. Such limitations, in addition to other shortcomings such as small screen size, low bandwidth, and limited power, lead to poor mobile Web browsing experience. We believe that a well-designed ubiquitous computing environment can narrow the gap in Web browsing experience by using context as default input for applications automatically.

In this paper, we use the zip code form filling as a case study to illustrate the usefulness of context in mobile Web. We implement a context-assisted proxy server, which automatically fills in the zip code part of a form for users. The proxy server examines the webpage requested from the mobile device, analyzes the input text which requires a zip code, and changes the INPUT to SELECT with OPTION filled in by context from the middleware which maintains the users' profiles, including locations and their visit histories. Preliminary results show that the context-assisted form filling approach is a good help when users browse the Web using mobile devices. This investigation demonstrates that context can indeed be used to reduce user's input effort.

A POPULAR ACTIVITY—USER INPUT

User input is very important during Web browsing. Users often need to enter the URI they are interested in to visit a particular webpage, fill out a form to check interested information, and write paragraphs to give comments or make blog entries. Form filling, as one type of user input, is quite popular in current webpages. A survey of the top 100 webpages from ALEXA.COM [1] reveals that 70% of these webpages contain forms such as search, email login, zip code, stock number, etc.

Personal data, which is part of user context, is one kind of information that many websites request, including users' interests, ages, and even their personal astrological signs, before providing some personalized service. For example, e-commerce is very popular nowadays and more and more customers go to websites to buy items. To make the purchase process more efficient, e-commerce sites normally use the user's interest as reference and keep track of his past purchases. One typical user context, location information, which can be obtained from a GPS receiver easily, is also desired by some websites. Ten of the top 100 websites provide weather information, schedules of nearby movie theaters, and the locations of restaurants based on zip code or city name.

The inclusion of WWW.WEATHER.COM and WWW.MAPQUEST.COM in the top 100 websites confirms that many users are indeed engaged in form-filling activities, since in these two websites users would get no information unless they complete the forms.

However, user input using mobile devices is often difficult. In order to provide high mobility, mobile devices are often small and have to use a keypad, soft keyboard, or stylus to substitute for the keyboard and mouse. The difficulty of

using such limited input facilities becomes one of the barriers for mobile Web applications.

CONTEXT IS HELPFUL

Based on the classification in [2], context can be categorized into physical context, user context, and computing context. We believe all of them are helpful for improving the user input process. The context used in Table 1 is just the user’s location and profile, which may be stored either in the mobile device or on the server.

Website type	Form items	Category
Search Engine	Keywords	N/A
News	Keyword, zip code, email, preference	Physical, user
Shopping	Address, state, full name, e-mail, telephone, bank account, preference	Physical, User
Organization	Keywords and e-mail	User
Blog	Login name and password, keywords	User
Tourism	Date, address, zip code, city, preference	Physical, user
Education	Keyword	N/A

Table 1. Items Requested by Different Website Categories

From Table 1 we can see that most of the form items requested by webpages can be filled in by context, which is not difficult to obtain. With the help of user context, the information could be easily filled in by the system, instead of through manual input. This observation motivates us to investigate the issues that may further the development of the ubiquitous Web.

ARCHITECTURE

Context, which can be stored in a middleware, could be shared by different applications, including user input for the ubiquitous Web. Letting the Web server directly communicate with a middleware may not be practical since it is expensive to modify each Web server in order to give every website context support. Moreover, giving websites too much personal information may cause privacy concerns.

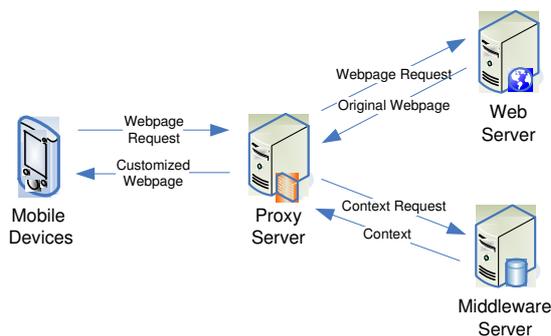


Figure 1. Proxy Enhanced Web Browsing

We believe a proxy-enhanced Web server is an appropriate solution. From Figure 1 we can see that all the webpage re-

quests are handled by a proxy server, which serves as a personalizing tool which takes advantage of the context from the middleware server and original webpages from the Web server. In this solution, the Web server does not need to be modified and the only configuration needed is proxy server selection in mobile devices.

CASE STUDY—CONTEXT-ASSISTED FORM FILLING

In our survey of the top 100 websites, we find that zip code filling is a popular activity. Therefore our context-assisted form filling system uses the zip code filling as a start. Moreover, in another survey about user context acceptance which was conducted in July 2005, we find that users are willing to provide their personal information in order to get context-assisted services. In order to understand how context works and the difficulty of implementing such systems, we implement a prototype system called “Context-Assisted Form Filling” which helps users automatically enter their zip code when they browse those webpages that request zip codes. We use a similar architecture as it is shown in Figure 1. The “mobile device suggestion” module at the proxy server consists of four components: request processing, preprocessing, adaptation, and page reconstruction, as shown in Figure 2. The system requests zip code information from a context middleware, changes element INPUT into element SELECT of the webpages transferred from original Web server, and returns the customized webpages back to users. By changing the interaction behavior from INPUT to SELECT, the input effort of users is reduced.

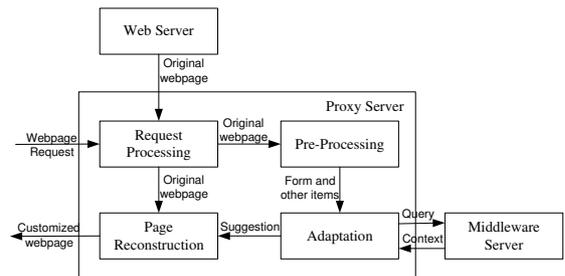


Figure 2. Mobile Device Suggestion Module

To obtain a larger number of webpages for our experiment, we used “ENTER ZIP CODE” as keyword in Google search in order to obtain a list of webpages which require zip code. We filtered out the dead links and incorrect query results, and took the first 18 webpages that requested users to input zip codes. Considering the popularity of the enter zip code pages, we also included the seven webpages which were in the top 100 webpages in our test sample. We got 34 forms in 25 different webpages. Our result is that 30 of those 34 forms were successfully modified using our context-assisted form-filling proxy server. One failure was caused by code written in a scripting language. A closer examination showed that the names of the text boxes in the other three forms that actually requested the zip code input were not meaningful (i.e., not related to zip code). Since we used the assumption that the webpage designers always gave meaningful names or values to the elements in forms, these three modification failures were not unexpected.

Automatic form filling is not limited to zip code filling. Some websites require users to enter their interested stock numbers. The proxy server could find the stock number the user is interested from the past search history. In this case, it is quite similar to zip code filling. With the help of a digital map, the proxy server can also convert the longitude and latitude information from GPS into a real address and fill in the form for users automatically.

DISCUSSIONS

Although we have demonstrated the feasibility of using context to improve the user's mobile Web browsing experience, there remain many challenges to widely deploy the solution. Some of the challenges need to be addressed first:

Meaningful label for HTML tags

Based on our experimental result of the context-assisted form filling system, we find that to give a meaningful name for an element is very important. If the website developer marks the INPUT name with a string containing "ZIP" or "LOCATION" the system successfully makes the modification. However, some website developers just use a simple string (e.g., "Q") to illustrate what the element is used for. This caused our prototype to fail. Although checking the text near the input element may help understanding the elements, the diversity of design approaches to produce a particular appearance makes accurate analysis difficult.

A best practice guide for website developers to name their elements based on the function is very useful in this case. Since our form filling system is only an example of using context, it is very helpful in general if website developers follow the best practice in making their intentions known.

Context request format

We expect there will be a diverse collection of context middleware to assist mobile Web browsing. It is important to specify a standard context request format. Otherwise the proxy servers and context servers would not know how communicate with each other.

Context itself also needs standardization. The W3C Mobile Web Initiative (MWI) [6] has published many working notes [4] and specifications [3] on how to describe a device, such as CCPP. The ubiquitous Web does not only "make Web access from a mobile device as simple, easy and convenient as Web access from a desktop device" [5], but also provides personalized services. Computing context is only one kind of context; additional specifications need to be proposed to describe user context and other physical context.

CONCLUSIONS

Ubiquitous computing enables the Web to extend personalized services based on context information provided by the sensors and user preferences. The ubiquitous Web is very attractive to users especially when they browse the Web via mobile devices despite device limitations. User input, which is a popular Web activities and currently difficult for mobile devices, can be helped by using context form middleware. However, there remain a lot of research issues to be solved

before wide deployment. Context format standardization and new Web best practice guidelines are not only helpful, but also essential.

REFERENCES

1. ALEXA. Alexa web search - top500. http://www.alexa.com/site/ds/top_500.
2. CHEN, G., AND KOTZ, D. A survey of context-aware mobile computing research. Tech. Rep. TR2000-381, Department of Computer Science, Dartmouth College, 2000.
3. W3C. Composite capability/preference profiles (cc/pp): Structure and vocabularies 1.0. <http://www.w3.org/TR/2004/REC-CCPP-struct-vocab-20040115/>.
4. W3C. Delivery context: Interfaces (dci) accessing static and dynamic properties. <http://www.w3.org/TR/2005/WD-DPF-20051111/>.
5. W3C. Mobile web best practices working group blog. <http://www.w3.org/2005/MWI/BPWG/>.
6. W3C. Mobile web initiative. <http://www.w3.org/Mobile/>.