

Personalized Portal for Wireless Devices

K. Maly

Department of Computer Science
Old Dominion University
Norfolk, VA 23529
757 683 4817

maly@cs.odu.edu

M. Zubair

Department of Computer Science
Old Dominion University
Norfolk, VA 23529
757 683 4817

zubair@cs.odu.edu

ABSTRACT

Most of the content available on the Web today is not suitable for Wireless devices such as mobile phones or PDAs, which are characterized by small sized display, small memory, low power, and low bandwidth access. The industry has developed Wireless Application Protocol (WAP), a standard that specifies the application framework and network protocols for wireless devices. In this paper, we propose a solution where users can build a personalized portal of traditional content on the Web and they can specify the relevant part of the content they would like to view on WAP enabled devices.

Keywords

Mobile devices, WAP, WML

1. INTRODUCTION

Most of the content available on the Web today is not suitable for Wireless devices such as mobile phones or PDAs, which are characterized by small sized display, small memory, low power, and low bandwidth access. The industry has developed Wireless Application Protocol (WAP), a standard that specifies the application framework and network protocols for wireless devices. The protocol stack defined in WAP optimizes standard Web protocols, such as HTTP, for use under the low bandwidth, high latency conditions often found in wireless networks. A number of enhancements to the session, transaction, security and transport layers provide HTTP functionality better suited to the wireless network environment. The WAP requires the content to be encoded in Wireless Markup Language (WML) to minimize the data sent to the wireless device and to minimize the computational energy required by the client to process the data.

The importance of this problem can not be overstated; almost all mobile phones sold today are WAP enabled and do have the small screen problem. There are phones and palmtops being developed by, for example, Nokia and Psion that have larger and high-resolution screens but due to their expense they are not expected to gain a significant market share over the next three years. Similarly, there are currently not more than a few percentage points of all web sites that are WAP enabled and it is not expected to reach double digit figures in the next three years.

It will be though a long time that all content on the Internet is WAP enabled. It requires a large investment for an organization to maintain dynamic content in two formats. Industry is addressing this by developing transcoding solutions, where the content in HTML is transformed to WML. Few examples under this category are: Transcoding solutions from IBM [1], UP.Link WAP Enhanced Services from Phone.com [2], and Spyglass Prism from OpenTV [3]. However, these solutions apply

generic filters while translating the content and consequently the transformed content may still contain information that a users do not want to view on their WAP enabled devices. In this paper, we propose a solution where a users can build a personalized portal of traditional content on the Web and they can specify the relevant part of the content they would like to view on WAP enabled devices.

2. ARCHITECTURE OF PERSONALIZED WIRELESS PORTAL SYSTEM (PWPS)

The basic approach we have taken is one of separating the process of identifying what a user wants to see on a web site and when that part of the web page is actually displayed on the WAP device for real-time access. That is, we want the WAP device user to indicate to the PWPS system, using a standard browser, what she would like to see and what part of a page she wants displayed on the WAP device. Let us illustrate this process by a sample scenario.

User Sets up the Wireless Portal using a Standard Browser:

For this part we assume that a user wants to set up her portal consisting of a single link. This link when clicked on a Wireless device will show her information she selected from the stock query result page that is typically returned on the NASDAQ web site when a user queries for a quote for specified stocks. Please note that here we are referring to the NASDAQ site (see <http://www.nasdaq.com>) that does not have WAP enabled content. 1. The user visits the PWPS web site and follows the instruction to access an arbitrary Web site, in our example the NASDAQ web site. Next she types in the appropriate input fields the stock symbols she wants to track. 2. After hitting submit, the user is now seeing the NASDAQ result page through a regular browser that shows all the fancy graphics, other links, other buttons and many details she is not interested in seeing on the WAP device. On the WAP device she only wants to see, for example, for each stock symbol she submitted, the current quote and the trade volume. She communicates this to the PWPS.

User Accesses the Wireless Portal using a WAP Enabled Device:

1. She visits the PWPS web site using her WAP device and views her portal, which consists of one link in our example. 2. The user clicks the link and she views the selected information from the query result page of the NASDAQ web site in real time. In other words, the user can track the desired information about stocks of her interest by one click on her WAP device.

During the personalized portal set-up phase the user downloads a "Controller Web page" from the PWPS web site. From this page a user starts another browser window to visit the NASDAQ web site. Next when she is viewing the query result page from the NASDAQ web site, she clicks a button on the "Controller Web Page" to indicate the PWPS that she is ready

for setting up the preference on the displayed Web page. Consequently, PWPS prepares a new page that shows the original page but with different background color added so that individual components on that page can be selected by the customizer. In addition a select button is added to each component. The customizer then selects, using the buttons provided, all the components that should be displayed on the WAP device and submits it back to the PWPS. The PWPS keeps the selection information and the original URL (including any query information that was added to the base URL). It then adds a new link (call it 'stock quotations') to the WML portal of the user that is stored on the WSP. When the WAP device user accesses the personal portal on the PWPS, the user will see this new link and can click 'stock quotations'. When the PWPS receives this WAP request, it will send the stored query to the associated URL and process the return page to filter out all information not to be returned. Next it translates the page into WML and returns it to the user.

3. PROTOTYPE IMPLEMENTATION

We have implemented the above strategy by concentrating on web sites where the highest level structures are tables. In our analysis we have found that 80% of all web sites fall into that category. We decided to show the user this structure of the page by color coding the table entries and do so recursively as our first attempt to solve the selection problem. That is, every element in a table is highlighted by a different color and that includes tables within tables. We then added a button to each entry, and let the user select what to display in the WAP device by clicking as many of the buttons and then submitting a done signal back to the WSP. We have also implemented the parsing of the return page and the selection of the indicated table entries using the stored profile of the appropriate PWPS. In our implementation we take this reduced html page and using standard translators produce a WML page that is returned to the WAP device. In our implementation the WAP device is simulated using the Phone.Com [2] simulator and the access of the personal portal is hardwired, that is for convenience we have only one PWPS that is accessed when the simulated WAP device is initiated. In Figure 1, we show the target Web page and Figure 2 shows the color-coded version that we are going to customize and add to our portal; it is from NASDAQ web site. In Figure 3 we show the mobile phone simulator displaying the selected content from the target Web page.

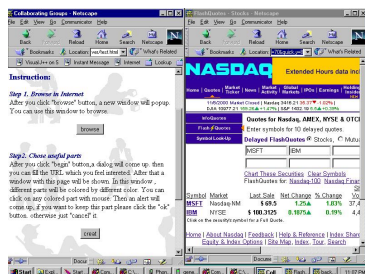


Figure 1. Selection from target page.

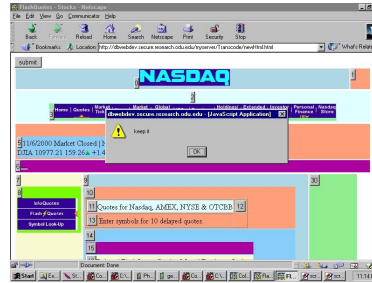


Figure 2. Color coded target page.

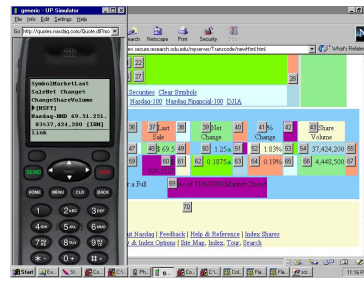


Figure 3. Phone simulator selection.

4. CONCLUSION

We strongly believe that at least for the next few years our approach for using WAP devices for interacting with the Web is a viable approach. Separating the process of capturing the type of interaction wanted and what to see from the process of actually doing it on the device has its advantages as well as disadvantages. The significant advantage in our approach is that general web sites do not have to adapt their site for WAP devices or even worse, maintain dual web sites. Secondly, we significantly reduce the traffic between the WAP device and the WSP which is usually where the bandwidth bottleneck is. Thirdly, we let the user choose what is displayed on the small screen of the WAP device.

5. REFERENCES

- [1] IBM Transcoding. <http://www-4.ibm.com/software/developer/library/transcoding.html>
- [2] Phone Com Uplink. <http://www.phone.com/products/uplink.html>
- [3] Spyglass Prism. <http://www.opentv.com/products/technologies/prism/applications.html>