

Association for Information Systems AIS Electronic Library (AISeL)

AMCIS 2006 Proceedings

Americas Conference on Information Systems
(AMCIS)

December 2006

Personalized and Tree-Navigation Interfaces for Mobile Web Applications

Boonlit Adipat

University of Maryland Baltimore County

Dongsong Zhang

University of Maryland Baltimore County

Follow this and additional works at: <http://aisel.aisnet.org/amcis2006>

Recommended Citation

Adipat, Boonlit and Zhang, Dongsong, "Personalized and Tree-Navigation Interfaces for Mobile Web Applications" (2006). *AMCIS 2006 Proceedings*. 349.

<http://aisel.aisnet.org/amcis2006/349>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2006 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Personalized and Tree-Navigation Interfaces for Mobile Web Applications

Boonlit Adipat

University of Maryland Baltimore County
Badipat1@umbc.edu

Dongsong Zhang

University of Maryland Baltimore County
zhangd@umbc.edu

ABSTRACT

Presently, there is an increasing demand of accessing Web from mobile handheld devices. However, due to the small screen of handheld devices, mobile users often suffer from the poor presentation that seriously hinders the usability of mobile applications. In addition, users have different information needs and preferences. They prefer to receive content of their interests in a preferred format. Therefore, it is important to adapt Web content to fit user preferences and provide users with more effective presentation on handheld devices. This paper proposes a new approach to addressing this problem. The approach automatically generates a tree-navigation interface for effective content browsing. It also takes users' interests and preferences into account while adapting the presentation for individuals. We describe our approach in detail and introduce Mobile Web, a system prototype that is implemented to demonstrate our approach.

Keywords

Content adaptation, personalization, mobile applications, interface design, human-computer interaction

INTRODUCTION

In recent years, the use of mobile handheld devices, such as personal digital assistants (PDAs) and cell phones, to access Web information has gained exponential popularity in business and normal daily life. These devices provide users with the potential of ubiquitous information access. There have been numerous emerging mobile applications that provide users with a variety of services such as daily news alert, restaurant and entertainment listings, and online shopping.

The small screen and low memory of handheld devices, however, present a critical challenge for taking advantage of their convenience and portability. Traditionally, there are two methods widely used for content browsing on handheld devices: scrolling and page splitting approaches. In the former approach, a user has to keep scrolling (up/down or left/right) to view the content, which is tedious and difficult to locate desirable information. In the latter approach, an original Web page is divided into a number of small pages. Users have to sequentially go through those small pages, which is likely to cause the problem of navigation loss (Qiu et al., 2004). Some studies (Baudisch et al., 2004; Chen et al., 2005) have proposed to first present users with a small-sized snapshot of a Web page. Users can later select a particular portion to zoom-in for more detail. However, this method is not practical with very small handheld devices such as cell phones because the size reduction usually causes the content in the snapshot illegible. Other studies generate a short version of a Web page (e.g., summaries and a list of topics). Users can quickly read through the short version and specify the content of their interests. However, generating the short version is so computation-intensive that could cause intolerable latency.

A promising solution to alleviate the problem of presenting large-sized information in a small screen is to provide an overview, a hierarchical structure of the overall content with links to separate sections in a Web page (Botafogo et al., 1992; Shneiderman, 1996). However, so far, there are a limited number of Web sites, which provide a version that contains an overview for handheld devices. Therefore, this paper proposes a new presentation method that adapts an original Web page to a mobile device and automatically creates a hierarchical view of its content. From the hierarchical view, users can specify topics that match their interests and later drill down for further detail.

Furthermore, most mobile Web applications do not tailor the content to satisfy personal information needs and preferences. For example, some users may want to read only political and business news, while others may be interested in entertainment and sports sections. Users have to read through irrelevant Web content until they find the desirable information. Several studies have found that presenting excessive information could decrease effectiveness of content browsing. Users may miss important information due to the confusion and the exhaustion of using working memory (Albers and Kim, 2000; Farhoomand and Drury, 2002). Therefore, it is important to deliver information that is personalized to match users' needs and preferences. In our approach, we take information about users' interests and preferences into consideration to tailor content and presentation for individuals.

The rest of the paper is organized as follows. In the next section, we introduce some related work on content presentation and personalization on mobile handheld devices. Then, we describe an algorithm for generating a hierarchical overview and personalizing content to fit personal interests in the following section. The final closing section discusses the future direction of this research.

RELATED WORK

Due to the influx of Web content, many approaches have been proposed to deal with the problem of information overload. Personalization has emerged as a promising solution to alleviate this problem. Personalization can be defined as any actions that a system adapts its behavior (e.g., tailor content) to match the needs of individuals according to user preference information (e.g., content of interest and preferred presentation formats). This information is normally stored in user profiles. There are different ways of creating individual user profiles, including direct user solicitation, explicit user feedback (requiring users to explicitly specify which information items are relevant), and implicit feedback (inferring users' interest by observing their information searching and/or browsing behavior). There has been extensive research on information personalization in desktop applications. However, relatively little personalization research has been done on mobile Web applications, although its importance has been well recognized.

In the past decade, there have been some HCI (Human-Computer Interaction) techniques proposed to adapt Web content to fit into the small screen size of mobile handheld devices. These techniques can be classified into four categories: presentation optimization, semantic conversion, zooming, and advanced focus & context visualization approaches (Adipat and Zhang, 2005).

Presentation optimization (PO) converts a multi-columned structure of a Web page into a single-columned structure. It fixes the horizontal width but changes the vertical length of a Web page. Users only need to scroll up/down (without left/right scrolling) to view content (Kaasinen et al., 2001). PO is simple but the frequent use of scrolling bar reduces reading speed. In addition, it increases users' cognitive load to remember information previously seen while they are scrolling (Jones et al., 1999). To minimize scrolling, some studies have proposed the page-splitting approach by dividing an original Web page into several small sub pages that can better fit the screen of handheld devices (Lum and Lau, 2002). This approach is intuitive because users can navigate content page by page as they are reading a book. However, after navigating through several pages, users could lose the position of where they are in the entire content (i.e., navigation loss). In addition, it is difficult for users to locate desirable information in specific pages.

Semantic conversion (SC) converts original content into a small-sized version by using heuristic and text summarization techniques. In the heuristic approach, some predefined rules are applied to parse HTML tags and extract content headers, which later are converted into hyperlinks. Users can click one of these links to switch to a new page and view the full content (Bickmore and Schilit, 1997). In the summarization approach, the original content is parsed and analyzed and important sentences are extracted as a summary based on linguistic cues. Then those summaries are presented to users (Buyukkokten et al., 2002). The SC approach enables users to quickly decide whether they are interested in the content by reading either the content headers or summaries. However, users have to take additional steps to retrieve and then view the original content, which causes longer delay. Furthermore, predefined rules in the heuristic approach are not flexible enough to be applicable to any Web documents, and summarization is a computation-intensive process that takes extensive processing time.

Zooming is a progressive process of expanding information into more/less details upon users' requests. Users are first presented with a small-sized snapshot of a Web page. Then, they can choose a specific area to view content in a greater size (Baudisch et al., 2004). Zooming is an intuitive approach. Users can easily migrate the way of Web interaction from desktop to handheld devices. However, with the small screen size of handheld devices, users may not be able to define the interested content area because the required size reduction typically makes the text on snapshot illegible.

Focus & context visualization (FCV) approach enables users to view information of their interest (i.e., the focus) in a segment of the screen area with the enlarged size of presentation, while showing other peripheral information (i.e., the context) in the surrounding area with the reduced granularity of detail (Leung and Apperley, 1994). With FCV, users can view information of interest (focus) without losing sight of peripheral information (context). However, a potential downside of this approach is that it is a computation-intensive process that may or may not be suitable for some handheld devices.

To address the above limitations, we propose an innovative approach that integrates the techniques of PO and SC with personalization. The method automatically adapts original content to fit right in the width of a screen and creates a hierarchical overview of an entire Web page on the fly. Our approach is different from all related work in three main aspects. First, it does not require intensive computation as summarization (SC) and focus & context visualization do. In addition, the overview is built upon the standard DOM tree (W3C, 2005), which is more flexible than building upon heuristic rules (SC).

Second, users can navigate through the overview and either expand specific sections for full content or collapse sections that seem irrelevant to their interests. Compared to PO methods, our approach reduces amount of scrolling. In addition, since the content is displayed in one page, it also reduces the problem of navigation loss. Third, we utilize user preference information (e.g., topics of interest and preferred presentation formats) to personalize the content and presentation. We will describe our approach in detail in the following sections.

TREE-NAVIGATION INTERFACES

Generally, when entering new Web pages on their desktops, users typically perform non-linear navigation rather than reading the content line by line. They scan through headings/sub-headings in the Web page to spot sections that they are interested. However, this is not effective for handheld devices due to the small size of screens that limit users from quick scanning through the entire Web content (Albers and Kim, 2000).

According to the visual information seeking mantra (Shneiderman, 1996), it is helpful to provide users with an overview of a document and enable them to drill down into particular sub-topics upon their preferences later. Therefore, we develop “Mobile Web”, a mobile application that automatically generates tree-navigation interfaces for any Web pages. Users can click at the bullet of a sub-section in the hierarchy to expand a tree and view more information. Figure 1 shows a few screenshots of a tree-navigation interface taken at HP iPAQ h4355 Pocket PC (with a 3.5” transfective screen and 64 K colors) via Pocket Screen Capture software. Other examples of tree-navigation interfaces are shown in the appendix (the last section of the paper).



Figure 1a.



Figure 1b.



Figure 1c.



Figure 1d.

Figure 1. A generated tree-navigation interface of the MSN index page

As shown in Figure 1b, when a user opens the MSN homepage, Mobile Web first displays an overview of the page. Then, he/she later can view more information about any particular sub-topics with a simple click to expand a section to the next level. In Figure 1c, the user clicks at the bullet of the first topic, Why does MSN look like this?, so Mobile Web expands the first section and displays information under the chosen topic. In Figure 1d, the user drills down through the tree and clicks to expand the topic, MSNBC.COM NEWS. With such a tree-navigation interface, we can avoid displaying the content of irrelevant sections of a web page and minimize the unnecessary page scrolling.

Mobile Web

Mobile Web is built upon a three-tier architecture. When a mobile client requests specific content (e.g. a Web page), an application server receives the request, searches for content from the Internet, adapts the content to fit a handheld device’s capability and a user’s preference, then delivers it back to the client. Due to the limitation of memory and processing capability, the client-side (i.e., a handheld device) includes only functions related to presentation such as displaying a tree-navigation interface and setting a presentation format (e.g., font size and media preferences). The server-side contains other computing-intensive content adaptation functions such as DOM tree generation – transforming an original Web page into a hierarchical tree-structured document, which is used to create a tree-navigation interface; and content personalization – determining content of interest based on user information (e.g., access history and content of previously visited Web pages). The Mobile Web system is designed to be wireless network independent. The architecture can be applied to many wireless

network infrastructures such as wireless LAN (WLAN) and 3G standards. We use JAVA servlet to create adaptation functions in the server-side and J2ME to display a tree-navigation interface in the client-side.

DOM Tree Generation

A tree-navigation interface is built upon the standard DOM (Document Object Model) tree, which creates a logical structure for HTML and XML documents in the form of tree-like representation (W3C, 2005). Each node in a DOM tree represents an object embedded in a Web document. Examples of objects are HTML/XML tags, images, and text. These objects are related to each other as parent or child nodes according to a beginning tag (e.g., <HTML>) and an ending tag (e.g., </HTML>). For example, any objects between a pair of BODY tags (e.g., <BODY> ... </BODY>) are considered as child nodes of the BODY node. The child nodes from the same parent will be at the same level in the hierarchy, as shown in Figure 2.

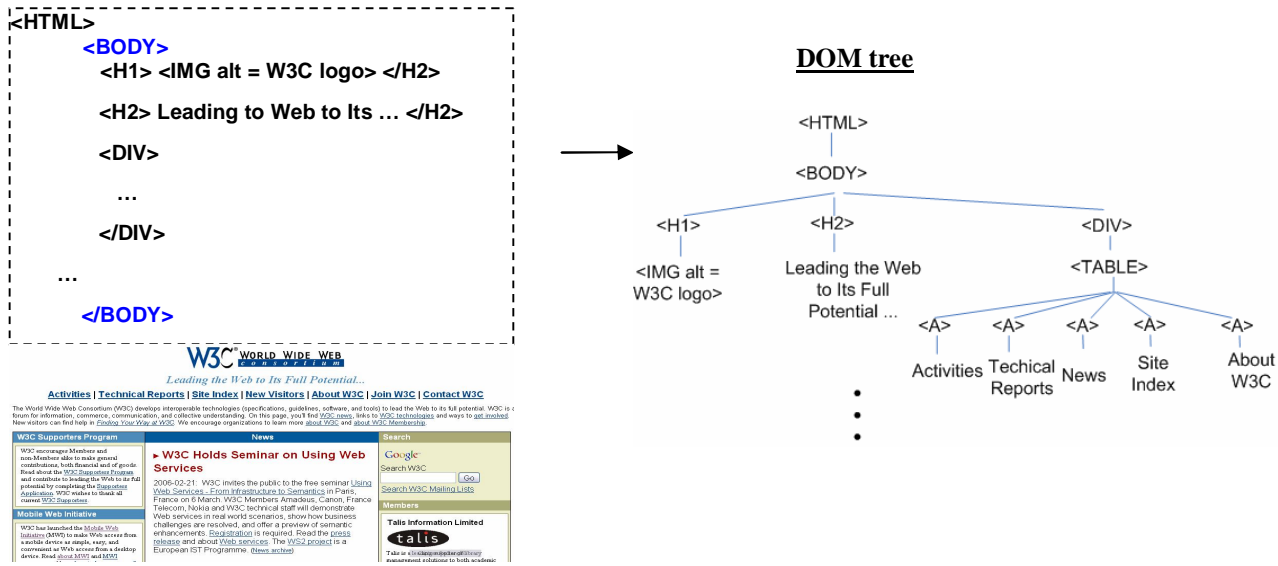


Figure 2. An example of a DOM tree of the W3C home page

Pruning the DOM Tree

Since users want to read only the content of a Web page, not HTML/XML markup tags, generating a tree-navigation interface mainly deals with analyzing the original source code of a Web page, removing tags, and replacing them with the nearest children that are content nodes (i.e., text or image). For example, in the News section of the W3C home page, our approach will remove tag nodes (e.g., <DIV> and <P>) and display only "News" as a parent node and all news headers as its children, as shown in Figure 3.

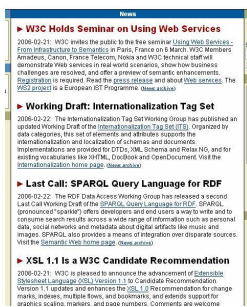


Figure 3a.

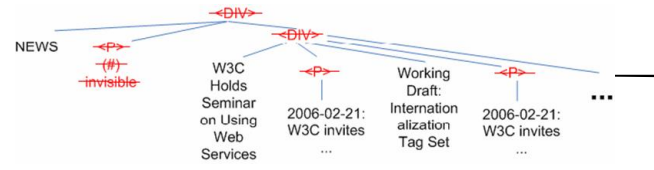


Figure 3b.

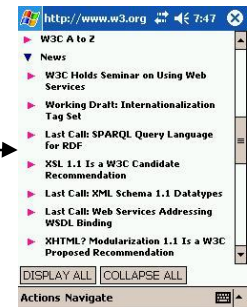


Figure 3c.

Figure 3. Generating a DOM tree navigation interface for the W3C homepage by removing tag nodes
Figure 3a. The News section of the W3C home page, Figure 3b. A DOM tree representation of the News section, and Figure 3c. A generated tree-navigation interface of the News section

We first use an HTML parser to generate an HTML DOM tree, which gives the position and information of each node in the DOM tree. Then, we traverse from <BODY> to its leaves and use the following algorithm to prune an original DOM tree. A new tree layout will contain only content nodes, but it still maintains logical relationship in the tree structure.

```

While (node != NULL){
  if (node != content){
    if (node = appearTag){
      // if the node is a tag node about appearance (e.g., font size/style and color)
      remove(node);
    }
    else if (node = organizeTag) {
      // if the node is a tag node about page format and organization (e.g., <DIV>, <P>, <BR>)
      remove(node);
      node.replace(content)
      tree.shiftUpLevel();
    }
    AssignTreeLevel(); // assign tree level numbers to nodes
  }
  else {
    // if the node is a content node, go to its child node
    node = node.nextChild();
  }
}

```

We first remove tag nodes that represent text/image appearance such as font, color, and hyperlink tags (e.g., , , <U>, <A>), as shown in Figure 4a). After removing the appearance tags, the tree will contain only content or tag nodes, which represent the format and organization of a Web page such as <P>,
, <DIV>. Then, we assign level numbers to the remaining nodes, as shown in Figure 4b). We traverse from the top node. If the node is a tag node, we remove the node, replace it with its child node that is a content node in the nearest lower level, and shift the entire lower level of the tree up, as shown in Figure 4b) and 4c). If the node is a content node, we traverse down to its children. The above process is repeated until all the nodes in the tree are content nodes. Mobile Web will first display only the highest level of a new tree structure. Users can later navigate to a lower level of any branch in the tree upon their preferences, as shown in Figure 4d).

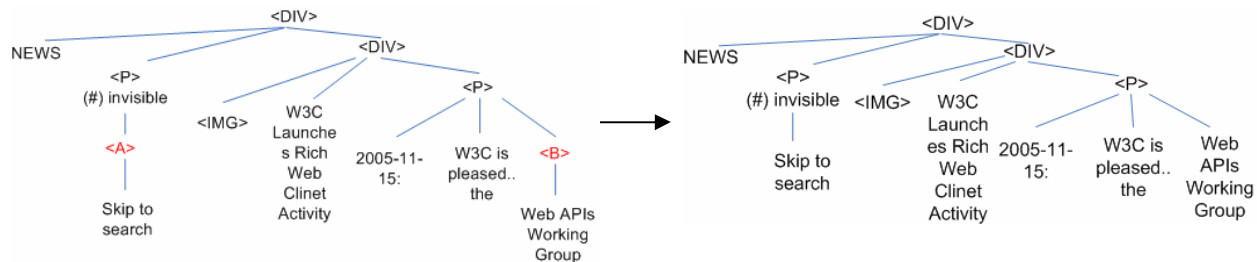


Figure 4a. Removing appearance tag nodes such as font, color and hyperlink tags

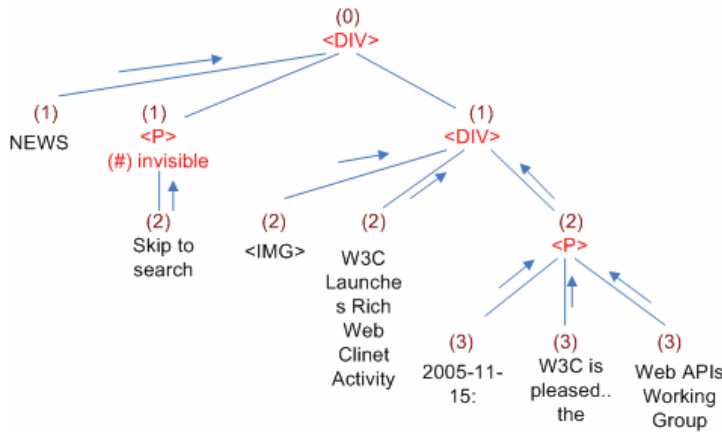


Figure 4b. Assigning level numbers to remaining nodes

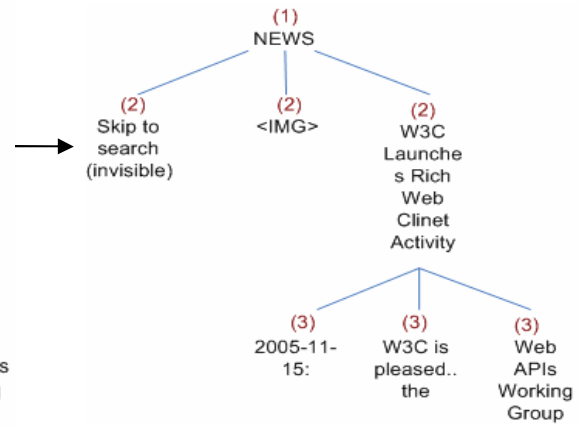


Figure 4c. Removing tag nodes and replacing them with content nodes

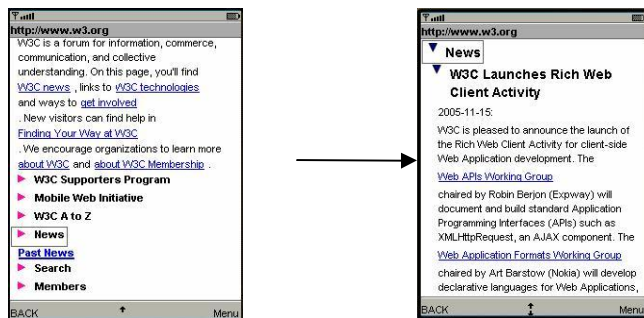
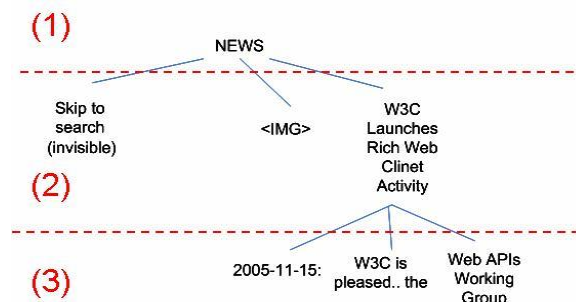


Figure 4d. Mobile Web first display nodes in the highest level of the tree layout. Users can later drill down to view full content



Currently, we draw a scope of generating tree-navigation approach with only static HTML. In addition, our algorithm does not take the presentation of frame and table into consideration. We are improving Mobile Web to include dynamic HTML (e.g., Java/VB script and CSS). However, we need to extend our algorithm for presenting frame and table.

PERSONALIZATION OF A TREE-NAVIGATION INTERFACE

Numerous studies have found that, in desktop, personalization provides several advantages. Some studies report that the use of information personalization saves users time to search for information, decreases number of errors, reduces the information overload problem, and improves users’ satisfaction (Blom and Monk, 2003; Quiroga et al., 2004). However, there are only a few mobile Web applications that help users personalize information based on their preferences. One of them is Daily Learner (Billsus et al., 2002), a mobile information system that personalizes news content. The system recommends a list of news articles that either receive high users’ ratings or contain content similar to an article that a user has read. However, Daily Learner is designed for only a specific web site, Yahoo! news. Besides, in Daily Learner, the adapted version of a Web page presented in a handheld device is so different from an original version presented in a desktop that a user cannot simply switch between the two platforms and continue their Web browsing.

In this research, we propose a personalization approach that can be applied to any Web pages. In addition, the version of a Web page presented in a handheld device still maintains the appearance to a degree that users can continue their Web browsing when switching from desktops to handheld devices.

Similar to desktop users, mobile users have different information needs and preferences on how information should be presented. For example, some users prefer to receive only text content to reduce transmission latency, while others do not mind spending longer time to download the content including high-resolution images and decorative features (e.g., colorful

backgrounds and graphics). Mobile Web utilizes this type of user information to tailor content and customize presentation to comply with users' preferences. This information is stored in a user profile.

User profile

User profiles are repositories that store information about users such as personal information interests and preferred presentation formats. A user profile in a mobile information system may consist of the following information:

- Demographic information such as age, gender, and background;
- Theme interests represented by keywords;
- Browsing history, such as the Web sites the user has visited, the time of last access, visiting frequencies, etc;
- Content presentation preferences such as preferred font size and media preferences (e.g., only text, text and image, and all media)
- Quality of Service (QoS) preferences; and
- Access privilege indicating what information the user can access

In a mobile computing environment, user profiles can be located at different locations, such as 1) a single centralized server as the only profile server, 2) different profile servers with duplicated or unduplicated user profiles, or 3) local mobile devices. Considering the cost of maintaining and accessing to user profiles, user profiles are normally stored on the server. Some researchers, however, have attempted to create and use user profiles on mobile devices. For example, Yau et al.(2003) develops a lightweight data mining approach that can be used on handheld devices. User profiles are generated by mobile devices, not by service providers. However, such client-side personalization is not practical due to the limited power and processing capability of mobile devices.

By taking this profile location issue into consideration, we use two user profiles in the Mobile Web system. One of them is stored in an application server. The other is in a local handheld device. The one in the application server is mainly used for content personalization. It contains information about individual users such as user name, occupations, topics of interest, and web access history. Mobile Web uses this information to select and tailor information. Another small-scaled user profile stored on the handheld device is mainly used for presentation personalization. It includes information about preferred presentation styles and formats (e.g., font size and color). Due to the constraints of handheld devices, Mobile Web uses this information only for adjusting the content presentation (e.g., display a Web page with the medium font size and without image). The information stored in these two user profiles is synchronized periodically. Table 1 and 2 illustrates the two user profiles in Mobile Web.

<u>User_Name</u>	<u>Password</u>	<u>Personal Interests</u>	<u>Occupation</u>	<u>Media Preference</u>	<u>Tolerating Latency(sec)</u>	<u>Device_ID</u>
Michael	Mike	Soccer, music, Mobile Tech	Stock broker	VDO, Image	20	4435629872
Tom	Thomas	Martial art, Stereo, Photos	General manager	Text, Image	10	4435896357

<u>Device_ID</u>	<u>URL</u>	<u>Frequency_requested</u>	<u>Latest_Access_Time</u>
4435629872	www.umbc.edu	2	28-MAY-04 02.20.48.287000 PM
4435629872	www.google.com	25	21-MAY-04 03.06.37.930000 PM
4435896357	www.ibm.com	19	23-JUN-04 08.46.32.631000 PM
4435896357	www.motorola.com	3	23-JUN-04 08.40.02.340000 PM

Table 1. A user profile in a server

<u>Mobile_ID</u>	<u>Personal_Interest</u>	<u>Media_Preference</u>	<u>Font_Size</u>	<u>Latest_Update_Time</u>
4435629872	Soccer, music, mobile tech	VDO, Image	2	28-MAY-04 02.20.48.287000 PM

Table 2. A user profile in a handheld device

User profiling

User profiling is the process of gathering information about users. This process is critical for the performance of content personalization (Quiroga et al., 2004). In Mobile Web, we use the combination of direct solicitation and implicit feedback approaches to collect user information.

Direct solicitation: In Mobile Web, users can specify their preferences of content (e.g., topics of interest) and presentation (e.g., font size and image removal) at any time of using their handheld devices. Such preferences, along with the description of interested topics (e.g., sport news, updated stock prices) stored in the client will be synchronized to the server immediately after users finish updating their profiles in the handheld devices.

Implicit feedback: Since users may change their interests and preferences over time, Mobile Web monitors individual users' Web browsing behaviors and automatically updates their profiles. Some of user behaviors that Mobile Web captures are presentation settings, URLs of Web sites that users have browsed, frequency of visits, and expanding/collapsing sections.

For example, if a user often changes font size, Mobile Web will keep the latest used font size in his/her profile and use it as the default font size for future content display. If a user always expands a specific branch of a tree-navigation interface such as the Entertainment section on the MSN home page, when he/she revisits the MSN site in the future, Mobile Web will automatically expand the Entertainment section and collapse other sections, and make it the current focus of this web page.

In addition, the Mobile Web system analyzes Web pages that users have visited and extracts keywords that can represent the users' interests. When users open a Web page and turn on the personalization function, Mobile Web will automatically expand and highlight sections that contain those keywords. For example, for a user who has "sport" as part of the interests, Mobile Web expands and highlights all sections that have the word "sport", as shown in Figure 5.



Figure 5. An example of a personalization feature: expanding and highlighting sections that contains the keyword of interest, "sport"

CONCLUSION AND FUTURE WORK

Nowadays, the use of mobile handheld devices to access Web is becoming popular. However, due to the small screen size of handheld devices, the presentation of the Web can easily become cluttered and cause difficulty for users to search for information they want. There have been some techniques proposed to present Web content on the small screen of mobile handheld devices, but those techniques still contain weakness, which hinders the usability of mobile Web applications. This paper attempts to make a contribution to this field by introducing a new approach to presenting Web content on mobile devices in the form of a tree-navigation interface. The approach also utilizes user information to customize content and presentation based on users' information interests and presentation preferences. We have described the algorithm for generating a tree-navigation interface and shown how to incorporate it with personalization. We strongly believe that the personalized and tree-navigation interfaces for Web content on handheld devices is a promising approach that can help alleviate the problem of the small screen of such devices while taking advantage of their convenience.

We have finished the first phase of prototype system development and testing. We are currently incorporating fisheye view (i.e., a technique that enables users to enlarge the content of interest, while reducing size of peripheral content) and summarization features. Next, we plan to add more services to Mobile Web such as a personalized multi-agent system and context-aware adaptation. Usability studies will be conducted to evaluate the effectiveness of individual features and the system as a whole.

APPENDIX

The following figures show examples of tree-navigation interfaces generated for different Web pages.

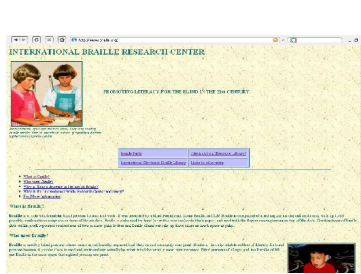


Figure 6a.



Figure 6b.



Figure 6c.

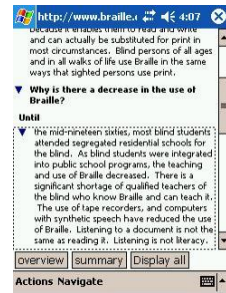


Figure 6d.

Figure 6. An example of a tree-navigation interface for the braille.org home page Figure 6a. The original Web page, Figure 6b. A list of topics in the braille.org home page, and Figure 6c and 6d. A user clicks on some topics to drill down for further details



Figure 7a.



Figure 7b.



Figure 7c.

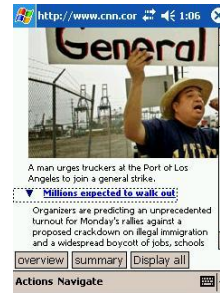


Figure 7d.

Figure 7. An example of a tree-navigation interface for the cnn.com index page Figure 7a. The original Web page, Figure 7b and 7c. A list of topics in the cnn.com index page, and Figure 7d. A user expands the tree to view more information



Figure 8a.

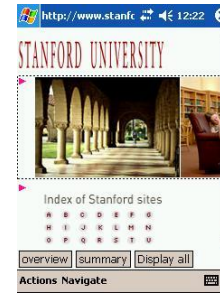


Figure 8b.

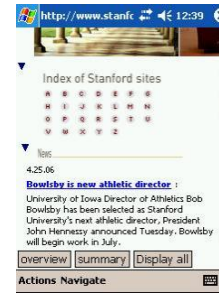


Figure 8c.



Figure 8d.

Figure 8. An example of a tree-navigation interface for the Stanford university home page Figure 8a. The original Web page, Figure 8b. A list of sections on the Stanford university home page, and Figure 8c and 8d. A user expands the tree to view more information

ACKNOWLEDGEMENTS

This research is partially supported by a research award from Google, Inc.

REFERENCES

1. Adipat, B., & Zhang, D. (2005) Interface Design for Mobile Applications, *Proceedings of the Eleventh Americas Conferences on Information Systems*, August 11th-14th, Omaha, Nebraska, USA.
2. Albers, M. J., & Kim, L. (2000) User Web Browsing Characteristics Using Palm Handhelds for Information Retrieval, *Proceedings of IEEE professional communication society international professional communication conference*, Cambridge, Massachusetts, USA, 125-135.
3. Baudisch, P., Xie, X., Wang, C., & Ma, W.-Y. (2004) Collapse-to-zoom: viewing web pages on small screen devices by interactively removing irrelevant content, *Proceedings of the 17th annual ACM symposium on User interface software and technology*, October 24th - 27th, Santa Fe, New Mexico, USA, 91-94.
4. Bickmore, T. B., & Schilit, B. N. (1997) Digester: Device-independent Access to the World Wide Web, *Proceedings of the 6th World Wide Web Conference*, April, 7-11, Santa Clara, CA, 665-663.
5. Billsus, D., Brunk, C. A., Evans, C., Gladish, B., & Pazzani, M. J. (2002). Adaptive Interfaces for Ubiquitous Web Access, *Communications of the ACM*, May, 2002, 45, 34-38.
6. Blom, J. O., & Monk, A. F. (2003) Theory of Personalization of Appearance: Why Users Personalize Their PCs and Mobile Phones, *Human-Computer Interaction*, 18, 193-228.
7. Botafogo, R. A., Rivlin, E., & Shneiderman, B. (1992) Structural Analysis of Hypertexts: Identifying Hierarchies and Useful Metrics, *ACM Transactions on Information Systems (TOIS)*, 10, 142-180.
8. Buyukkokten, O., Kaljuvee, O., Garcia-Molina, H., Paepcke, A., & Winograd, T. (2002) Efficient Web Browsing on Handheld Devices Using Page and Form Summarization, *ACM Transactions on Information Systems*, 20,1, 82-115.
9. Chen, Y., Xie, X., Ma, W.-Y., & Zhang, H.-J. (2005). Adapting Web Pages for Small-Screen Devices, *IEEE Internet computing*, January-February, 9, 50-56.
10. Farhoomand, A. F., & Drury, D. H. (2002). Managerial Information Overload, *Communications of the ACM*, October, 45, 127-131.
11. Jones, M., Marsden, G., Mohd-Nasir, N., & Buchanan, G. (1999) A site based outliner for small screen web access, *Proceedings of the 8th World Wide Web Conference*, May 11-14, Toronto, Canada, 156-157.
12. Kaasinen, E., Aaltonen, M., Kolari, J., Melakoski, S., & Laako, T. (2001) Two approaches to bringing Internet services to WAP devices, *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 33,6, 231-246.
13. Leung, Y. K., & Apperley, M. D. (1994) A Review and Taxonomy of Distortion Oriented Presentation Techniques., *ACM Transactions on Computer-Human Interaction*, 1,2, 126-160.
14. Lum, W. Y., & Lau, F. C. M. (2002) A Context-Aware Decision Engine for Content Adaptation, *Pervasive Computing*, July - September 2002, 41-49.
15. Qiu, M. K., Zhang, K., & Huang, M. (2004) An Empirical Study of Web Interface Design on Small Display Devices, *Proceedings of 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, September 20-24, Beijing, China, 29-35.
16. Quiroga, L. M., Crosby, M. E., & Iding, M. K. (2004) Reducing Cognitive Load, *Proceedings of the 37th Hawaii International Conference on System Sciences 2004*, Jan 5-8, Hawaii, 131-139.
17. Shneiderman, B. (1996) The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations, *Proceedings of the IEEE Symposium on Visual Languages*, September 3-6, Washington, DC, 336--343.
18. W3C. (2005). Document Object Model (DOM), January 19, Retrieved Febuary 27, 2005, from <http://www.w3.org/DOM/>
19. Yau, S. S., Liu, H., Huang, D., & Yao, Y. (2003) Situation-Aware Personalized Information Retrieval for Mobile Internet, *Proceedings of the 27th Annual International Computer Software and Applications Conference (COMPSAC'03)*, Nov. 3-6, 639-644.