Personalization and Visualization on Handheld Devices

Dongsong Zhang, George Karabatis, Zhiyuan Chen, Boonlit Adipat, Liwei Dai, Zhenxue Zhang, Yu Wang

Department of Information Systems, University of Maryland, Baltimore County (UMBC)
1000 Hilltop Circle, Baltimore, MD 21250, USA
{zhangd,georgek,zhchen,badipat1,liwei1,zzhang3,wangyu1}@umbc.edu

ABSTRACT

The small screen size of handheld mobile devices poses an inherent problem in visualizing data: very often it is too difficult and unpleasant to navigate through the plethora of presented information. This paper presents a novel approach to personalized and adaptive content presentation for handheld devices, which has been implemented in a mobile financial application system based on a 3-tier architecture. The approach is independent of wireless networks and mobile devices. It utilizes a combination of user profiling, data clustering, and visualization techniques (fisheye and semantic zooming), enhancing the understandability of the data and improving the usability of the device.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *clustering, information filtering, relevance feedback, search process, selection process.*
H.5.2 [**Information Interfaces and Presentation**]: User Interfaces – *graphical user interfaces, user interface management systems.*

General Terms

Design.

Keywords

Mobile devices, PDA, user profiles, visualization, clustering.

1. INTRODUCTION

Many wireless applications have been developed for mobile handheld devices, including daily news alerts, classified mobile advertising, restaurant and entertainment listings, wireless Web portals, mobile commerce (m-commerce) applications, etc.

The ability to communicate from virtually anywhere at any time and the convergence of Web and wireless technologies offer an unprecedented level of flexibility and convenience for ubiquitous Web access through mobile devices. The unique constraints of wireless networks and mobile devices, however, present a number of critical challenges for taking advantage of the convenience of mobile devices and ubiquitous information access. Some of them are [13]:

Navigation. Mobile handheld devices are featured with small screens, possibly making presentation of most Web pages or data aesthetically unpleasant, un-navigable, and in the worst case, completely illegible [3]. Therefore, it is essential to find better ways to present information on a mobile device.

Bandwidth. Although network technology advances rapidly and network bandwidth increases, in practice wireless network bandwidth is still limited and expensive. In addition, information overload exacerbates this problem. It requires that information delivered to mobile users should be relevant and tailored to satisfy their individual needs.

Disconnections. Users of wireless networks experiencing disconnections cannot access information and perform any further operations associated with the server.

This paper addresses mainly the navigation and visualization issues in a mobile financial application for users interested in finding information about stocks and mutual funds. Our contributions are:

- A prototype system is developed based on a three-tier architecture to obtain the latest relevant stock or mutual fund information from specific Web sources, deliver it to a mobile device with two-level caching, and present it to users in an effective manner.

- User profiling techniques are utilized to create specific information filters and streamline the amount of data sent to the mobile clients.

- Visualization techniques are incorporated to assist users' navigation and comprehension of information presented on the handheld device. A fish-eye view coupled with a semantic zooming visualization technique are used to demonstrate the trend and other properties of a stock. In addition, a hierarchical data clustering technique is also employed to help users navigate a large number of stocks or funds.

The unique combination of the above visualization techniques in conjunction with user profiling presents a novel approach to assisting users to view large amounts of information in a meaningful way through the small screen of a handheld device. The rest of the paper is organized as follows. Section 2 introduces previous related work. The prototype mobile information system architecture and major technical components are discussed in Section 3. Finally, the paper concludes in Section 4.

2. RELATED WORK

Many different approaches have been proposed to deal with the problem of information overload. Personalization is commonly used to alleviate this problem. Personalization can be defined as a

set of actions that a system takes to provide customized services, which match the needs of a particular user or a set of users according to user information (e.g., interests, preferences) collected through user interaction with the system [4, 8]. Personalization is emerging as a promising solution which enables users to quickly obtain desirable information [1, 11].

There are different ways of creating individual user profiles, including direct user solicitation, explicit user feedback (requiring users to explicitly specify which delivered documents are relevant), and implicit feedback (inferring users' interest by observing their information searching and/or browsing activities). The profiles can contain not only demographic information about users, but also their information needs and personal preferences required for content customization. These profiles can be further processed and clustered to form user groups based on shared interest. As a result, a mobile information system can use push proxies to deliver certain information automatically to a group of users with common interests. There has been extensive research on information personalization in desktop applications. However, so far, relatively little personalization research has been done in mobile information systems, although its importance has been well recognized.

There have been studies on how to better present information on handheld devices with small screen size [12]. In general, the

In [7], a labeled, hierarchical category navigation tree structure is dynamically generated for a set of records. The model uses the tree structure to show summaries of content with different granularities. Users can determine whether a specific category is relevant or not by simply examining its label and explore only the relevant categories, thereby reducing the information overload problem. However, the generated tree is fixed, and the tree creation algorithm does not take into account skewness and correlations in data. In this paper we use an interactive hierarchical clustering technique to help users navigate through a large number of records about stocks and mutual funds. Compared to [7], our clustering approach is interactive and can handle skewness and correlations in data.

Another line of research on information presentation on mobile handheld devices focuses on visualization techniques. *Focus plus context* is a visualization technique that enables users to focus on an important part of the content while keeping a contextual view. Many *focus plus context* techniques using different distortion/suppression approaches have been developed, such as the fisheye technique [9] and flip zooming technique [5]. Several of them are applicable to mobile devices, such as the DateLens (a fisheye-enabled calendar for mobile devices) [2]. In this research, we have adopted a combination of the fisheye and semantic zooming techniques to view the trend of a stock.
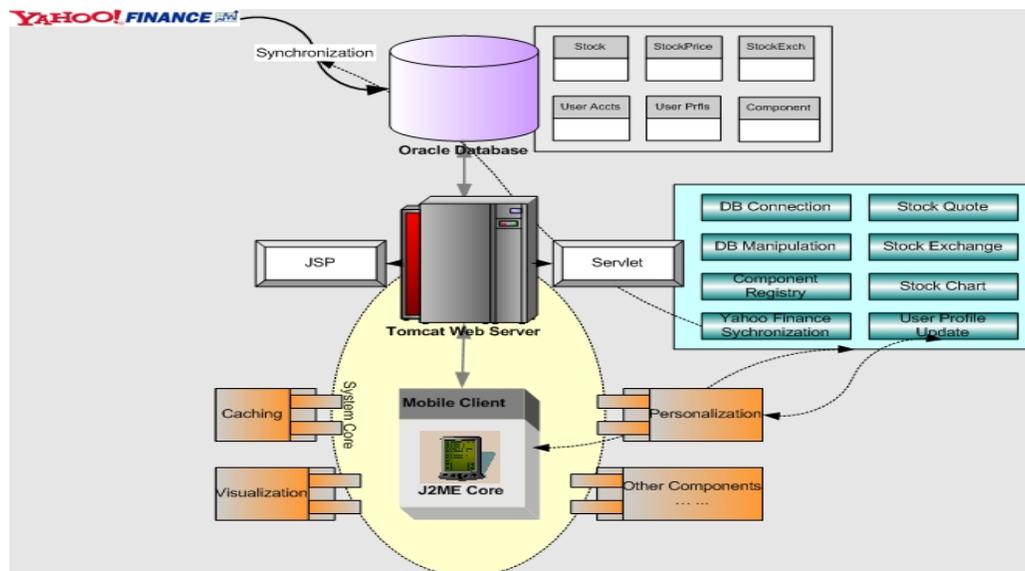

Figure 1. The three-tier architecture of the system

approaches can be classified into two categories: content summarization and data visualization. In the first type of approach, rather than displaying the entire content to a user, only a summary is presented. The user can determine whether a particular document is relevant and interesting based on its summary and whether s/he wants to see the entire content, thus, avoiding unnecessary browsing of irrelevant information. For example, in the work of Buyukkokten *et al.* [6], a multi-level abstraction of textual Web documents is generated, ranging from original Web documents to summaries and keywords of documents. Users can retrieve the entire content after viewing the summaries or keywords. However, it is not clear how to apply this technique to a set of data records (e.g., information of a large number of mutual funds or stocks).

3. A PROTOTYPE SYSTEM FOR MOBILE FINANCIAL APPLICATIONS

3.1 System Architecture

The dynamic fluctuations of the stock market are considered very important for investors to view the latest stock information and monitor the trends of stocks anywhere, anytime. We have developed a prototype system for stock querying and trend monitoring using mobile handheld devices. Our system obtains stock prices from the Yahoo! Finance online stock database and other online sources, filters and organizes them based on user preferences stored in user profiles, and then displays them on a handheld device. User profiles are dynamically and automatically updated based on users' stock searching and browsing behavior.

As shown in Figure 1, the prototype system was developed based on a three-tier architecture: the Oracle database as the back-end of the system, the Web and application server (Apache Tomcat 5.0.19 Server and J2EE Servlet) as the middle-tier, and the mobile client (with Java MIDlets – applets for mobile devices) as the client-tier.

We chose to use the Java™ programming language to implement this prototype system because of its portability and cross-platform compatibility. The database server maintains stock prices obtained from predefined online sources by a Java Servlet on a daily basis. The middle tier, the application server, is responsible for receiving, processing, and replying to client requests. On the client side, we utilized the Java 2 Platform, Micro Edition (J2ME) Wireless Toolkit 2.0 on a PalmOS emulator to implement a thin-client, offloading major computational workload to the application server. J2ME adapts to and addresses: 1) low quality wireless communication ─ slow and unreliable networks; 2) mobility and ubiquitous integration; 3) portability or pervasive devices; and 4) limited requirements of device hardware. The MIDlets on the client devices serve as gateways to servlets hosted on the middle tier. For example, there is a MIDlet called StockQuoteMIDlet, which gets input (e.g., symbol and time) from users, calls a relevant servlet (StockQuoteServlet), receives a response, and displays results on the end-user's handheld device.

3.2 System Functionality

There are five major functions in the prototype system implemented either as applications or MIDlets depending on the tier they reside. The *Stock Synchronization* function periodically downloads the current stock and mutual fund data using the Yahoo! Finance API and updates the Oracle stock database; The *Stock Quote* function allows users to request prices of specific stocks or funds within a time period. Users can either issue a query or select certain stocks from a personalized list that is automatically created based on the past user history; The *Stock Chart* function generates bar charts dynamically for stock or mutual fund comparisons; The *Stock Trend View* function adopts the fisheye visualization and semantic zooming techniques to enable users to view the trend of any stock or fund at multiple granularity levels; and the *Stock and Mutual fund Navigation* uses a hierarchical clustering technique to help users navigate through a large number of stocks or mutual funds and select the ones they

want to look for more details. Next we describe the visualization techniques we used for the stock chart and stock trend view functions, personalization techniques for caching and customization, and clustering techniques for the navigation function.
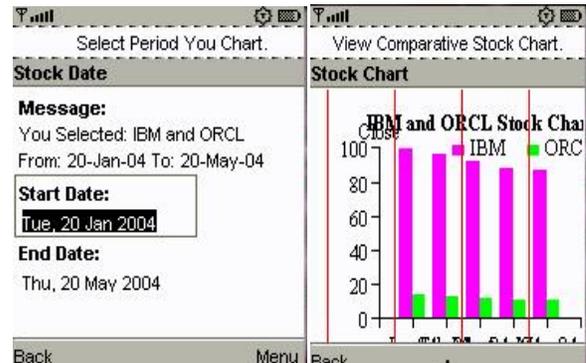

Figure 2. Bar chart visualization for stock comparison

3.2.1 Viewing Stock Information Using Bar Charts

To better present stock information on small-screen devices, appropriate visualization techniques should be used. Our prototype can present stock data to users via bar charts. This function allows users to view information on any single stock, or compare multiple stocks within a given time period in a single chart. Bar charts are dynamically generated by Servlets based on the stock quotes extracted from the database. An example request for stock comparison between IBM and ORCL and the generated bar chart are illustrated in Figure 2.

3.2.2 Using Semantic Zoom-in/out Capability

Due to the limited screen size, large amounts of data presented on a small screen may require significant cognitive effort from users; sometimes they may be illegible or make no sense to users (e.g., Figure 3(a)). To alleviate this problem, our system adopts the fisheye technique, enhanced by a semantic zooming technique, to present stock prices and their trends. The fisheye technique is used when there is too much information to fit in one screen, but
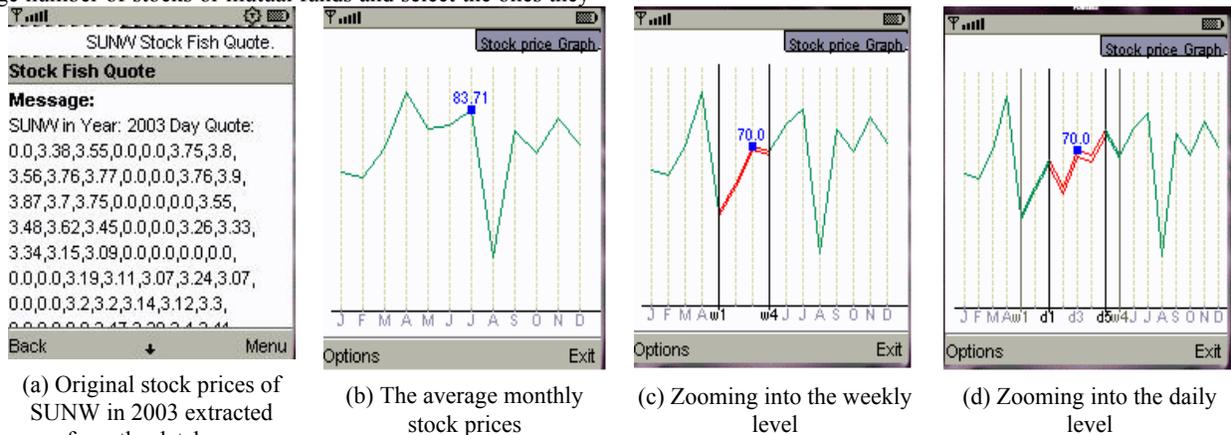

(a) Original stock prices of SUNW in 2003 extracted from the database

(b) The average monthly stock prices

(c) Zooming into the weekly level

(d) Zooming into the daily level

Figure 3. A view of stock prices using the fisheye and semantic zooming techniques

the user still wants to see the details while keeping its context. Identifying the trend, outliers, and the value of data with a time dimension is very important but difficult to display on a small screen, especially when details are essential. Simply presenting all stock prices as text over a long period (e.g., 1 year) is not effective or even feasible.

We have applied several *focus plus context* approaches for 2-D visualization. That is, in a high-level view (e.g., monthly view), only summaries of lower level data are presented (e.g., monthly average stock prices). Users can drill down to a lower level of a specific data point (such as weekly view of a month or daily view of a week) based on their interest. In this case, the focus area (i.e., the focused month) is expanded and more detailed information (e.g., weekly or daily average prices) is displayed, while the context area — peripheral data points (e.g., average prices of other months) are still visible but reduced in size.

In our system, the fisheye technique is used on the generated line charts, which is enhanced with the capability of zooming in and zooming out along a three-level hierarchy: Month→Week→Day enabling a 'drill-down' analysis. For example, assume that a user is interested in the stock information of Sun Microsystems 'SUNW' during a specific year. The system retrieves the stock data and generates a line chart. Figure 3(b) shows 12 monthly average stock prices (close prices) for SUNW in 2003. In Figure 3(c), the red line is the result of week zoom-in operation, displaying average weekly prices in May 2003. Similarly, Figure 3(d) drills down to a daily level in a particular week of May, while still displaying all the contextual data (adjacent weeks and other months). A single focus data point can also be highlighted with more detailed information (e.g., stock price value). Once the user moves the highlighted point out of the current focus level, the system assumes that the user's interest has changed and the view automatically zooms out to a higher level, enabling a smooth visual change.

3.2.3 Personalization and Customization
In the context of financial investments like purchasing stocks, it is quite beneficial to users if the latest updates of stocks would be automatically 'pushed' to them without frequent requests. Therefore, a personalization component is integrated in our system. It resides on both client and middle tiers and utilizes user profiles. Each user has a profile that contains not only demographic information and stock interest of a user, but also recent stock history of searching and viewing behavior (e.g., query logs). Two copies of user profiles are stored separately in two databases, one located in the handheld device, and the other one in the server, respectively. They synchronize with each other when a connection is established. Based on these profiles, the system is capable of providing the following personalized services to the users:

- Stock update alerts: the system periodically examines the changes stock values on Yahoo! Finance and synchronizes its stock database. An identified stock update will be automatically forwarded to certain users based on their profiles. Such an alert service keeps users up-to-date without having to send explicit requests, thus accelerating the information acquisition and reducing network traffic.

- Caching data: Because a wireless network is unreliable, the system caches the most relevant and/or recent data on local devices, increasing data availability even when devices are disconnected from the server.
- Automatic profile updates: Stock information is displayed on mobile devices in a decreasing order of users' interest, implied from their past access history. If a user has neither requested for nor viewed prices of a particular stock for a pre-defined time period, this stock is automatically removed from the user's interest list.
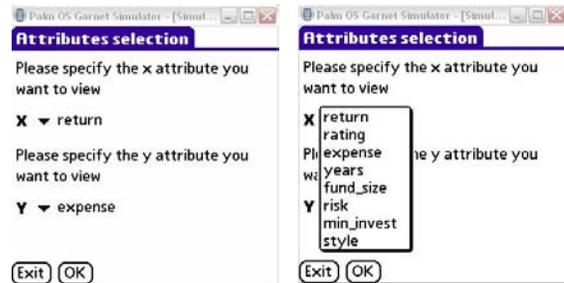

Figure 4. Attribute selection screens for a user to select two attributes as clustering criteria.

3.2.4 Navigation Using Hierarchical Clustering
It may be difficult for some users to identify stocks or mutual funds especially when they are searching for the first time. Moreover, displaying a large number of stocks or mutual funds on a small screen exacerbates the problem. In our system, we use a hierarchical clustering technique to group similar stocks/funds into clusters, and organize them into a hierarchy such that users can visualize these clusters, navigate through them, and decide which cluster(s) they want to explore further. This approach is extremely scalable because the height of the hierarchy is logarithmic to the number of data points. Unlike previous studies [7], our approach generates clusters interactively, and can better handle skewness and correlations in data because clustering
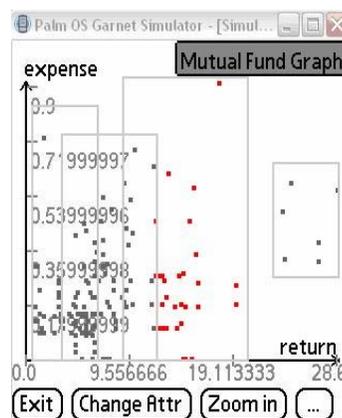

Figure 5. Mutual fund clusters (rectangles) based on *return* and *expense* attributes.

methods are robust to skewness and correlations in data [10].

The navigation process is as follows. Users first select the attributes they want to use to cluster the stocks or funds (typically two for ease of visualization – see Figure 4). On the middle tier server, a hierarchical clustering algorithm is then executed to cluster stocks using those attributes (e.g., return and expense), and a hierarchy of clusters is generated. Clusters at the top level of the hierarchy are then displayed as shown in Figure 5. Users can either tap a cluster and zoom into sub-clusters of the selected cluster, or start over and use two different attributes for clustering (as shown in Figure 6), or zoom out to a higher level cluster. In the first case, users can reach a point where a cluster contains only a small number of stocks or funds and then examine their details (which can fit in a small screen). In the second case (Figure 6), we use a linking technique to show previously selected stocks or funds in the new clusters. Previously selected funds (presumably satisfying user criteria for the *return* and *expense* attributes) are highlighted such that users can choose among these funds that also satisfy *rating* and *risk* criteria. As a result, users can visualize information in a tidy, precise, and efficient manner and easily navigate up and down through the cluster hierarchy.
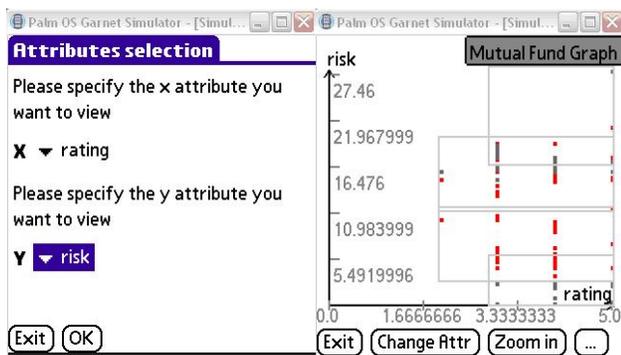

Figure 6. Linking technique between old (*return* and *expense*) and new (*risk* and *rating*) clustering attributes.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we addressed the problem of visualizing information on small screens of handheld devices, especially when significant amounts of data are to be displayed. Our approach integrates personalization and visualization techniques. We described our prototype system, which was built based on a three-tier architecture, and presented our visualization techniques with semantic zooming and clustering to display mostly numeric financial data. Personal user profiles are used to filter out unnecessary information from being transferred to the handheld devices. A combination of fish-eye and semantic zooming techniques enables users to visualize large amounts of data on a small screen. In addition, hierarchical clustering assists users to navigate through a plethora of financial data points organized in a hierarchy of clusters, enabling faster data navigation and decision making. We implemented our prototype in J2ME for platform independence and increased portability regardless of the device characteristics. In addition, the bar chart visualization component of our application also runs on smart phones. Our future plans

include user evaluation of our prototype in terms of usability and effectiveness, as well as the exploitation of the applicability of our approach to other domains.

5. REFERENCES

1. Anderson, C.R., Domingos, P. and Weld, D.S. Personalizing Web Sites for Mobile Users. In *WWW*, (Hong Kong, 2001).
2. Bederson, B., Clamage, A., Czerwinski, M.P. and Robertson, G.G. DateLens: A Fisheye Calendar Interface for PDAs. *ACM Transaction on Computer-Human Interaction 11*, 1 (2004), 90-119.
3. Bickmore, T. and Schilit, B. Digestor: Device-independent Access to the World Wide Web. In *6th WWW Conference*, (Santa Clara, California, 1997), 665-673.
4. Billsus, D. and Pazzani, M. A Personal News Agent that Talks, Learns and Explains. In *Third International Conference on Autonomous Agents (Agents '99)*, (Seattle, Washington, 1999), 268-275.
5. Bjork, S., Homquist, L.E. and Redstrom, J. A Framework for Focus+Context Visualization. In *IEEE Symposium on Information Visualization*, (San Francisco, California, 1999).
6. Buyukkokten, O., Garcia-Molina, H. and Paepcke, A. Seeing the whole in parts: text summarization for web browsing on handheld devices. In *WWW*, (2001), 652-662.
7. Chakrabarti, K., Chaudhuri, S. and Hwang, S.-W. Automatic Categorization of Query Results. In *ACM SIGMOD Conference*, (2004), 755-766.
8. Eirinaki, M. and Vazirgiannis, M. Web Mining for Web Personalization. *ACM Transactions on Internet Technology 3*, 1 (2003), 1-27.
9. Furnas, G.W. Generalized fisheye views. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, (Boston, Massachusetts, United States, 1986), 16-23.
10. Han, J. and Kamber, M. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2000.
11. Joachims, T., Freitag, D. and Mitchell, T. WebWatcher: A tour guide for the World Wide Web. In *International Joint Conference in AI (IJCAI 97)*, (Aichi, Japan, 1997).
12. Qiu, M., Zhang, K. and Huang, M. An Empirical Study of Web Interface Design on Small Display Devices. In *IEEE/WIC/ACM International Conference on Web Intelligence*, (Beijing, China, 2004), 29-35.
13. Zhang, D. Delivery of personalized and adaptive content to mobile devices: a framework and enabling technology. *Communications of AIS 12* (2003), 183-202.