An Application Streaming Service for Mobile Handheld Devices

Joeng Kim, Ricardo A. Baratto and Jason Nieh
Department of Computer Science
Columbia University, New York, NY, USA
{jk2438, ricardo, nieh}@cs.columbia.edu

Abstract—
Mobile handheld devices such as PDAs and smartphones are increasingly being used by service providers to deliver application functionality similar to that found in traditional desktop computing environments. However, these handheld applications can be quite slow and often lack important functionality compared to their desktop counterparts. We have developed PASSPORT, (PDA Application Streaming Service PORTal) a thin-client solution that leverages more powerful servers to run full-function desktop applications and then simply stream screen updates to the PDA for display. PASSPORT uses server-side screen scaling to provide high-fidelity display and seamless mobility across a broad range of different clients and screen sizes, including both portrait and landscape viewing modes. PASSPORT also leverages existing PDA control buttons to improve system usability and maximize available screen resolution for application display. We have implemented PASSPORT on Windows PDAs and demonstrate that it can provide superior application performance and functionality compared to the traditional approach of running applications directly on handheld devices.

I. Introduction

The increasing ubiquity of wireless networks and decreasing cost of hardware is fueling a proliferation of mobile wireless handheld devices, including wireless Personal Digital Assistants (PDA) and integrated PDA/cell phone devices. These devices are enabling new forms of mobile computing and communication. Service providers are leveraging these devices to deliver general application functionality similar to what is found in traditional desktop computing environments, including web browsing, email, video, music, financial planning, and personal information management.

These devices are typically used by running applications locally on them. Although native applications exist for PDAs, many of them deliver subpar performance and have a much smaller feature set and more limited functionality than their desktop counterparts. This fundamental problem arises for two reasons. First, since PDAs have a completely different hardware/software environment from traditional desktop computers, applications need to be rewritten and customized, duplicating development costs. Because the desktop application market is larger and more mature, most development effort generally ends up being spent on desktop applications, resulting in greater functionality and performance than their PDA counterparts. Second, PDAs have a more resource constrained environment than traditional desktop computers to provide a smaller form factor and longer battery life. Desktop applications are often large and complex and unable to run on a PDA. Instead, developers are forced to significantly strip down these applications to provide usable PDA applications, thereby crippling PDA application functionality.

To address these problems, we propose an alternative solution for delivering application services on mobile handheld devices by using thin-client computing. In this model, handheld devices communicate over a network with a server using a remote display protocol. The protocol enables graphical displays to be virtualized and served across a network to a handheld client device, while application logic is executed on the server. Using the protocol, the client transmits user input to the server, and the server returns screen updates of the applications from the server to the client. Mobile handheld devices become simple stateless clients that leverage the remote server capabilities to execute applications.

A long standing limitation of thin clients has been that they were not designed for PDA usage, were ineffective at providing seamless mobility across different display sizes, and could not effectively support more display-intensive applications. However, we have recently developed pTHINC [9], a new thin-client architecture that overcomes these previous limitations in the context of mobile web applications. Our previous work has shown that pTHINC can significantly improve web application usability, functionality, and performance on mobile handheld devices.

Using pTHINC, we have developed PASSPORT (Pda Application Streaming Service PORTal), a thin-client architecture that can be used by service providers to deliver general application services on mobile handheld devices. PASSPORT virtualizes and resizes the display on the server to efficiently deliver high-fidelity screen updates to a broad range of different clients, screen sizes, and screen orientations, including both portrait and landscape viewing modes. This enables PASSPORT to provide the same persistent session across different client devices. We have implemented PASSPORT and demonstrated that it works transparently with existing applications, window systems, and operating systems, and does not require modifying, recompiling, or relinking existing software.

PASSPORT provides several important benefits for supporting application services on mobile handheld devices. First, standard desktop applications can be utilized on PDAs without rewriting or adapting applications to execute on a PDA, reducing development costs and leveraging existing software

IEEE International Conference on Services Computing (SCC'06)
0-7695-2670-5/06 $20.00 © 2006 IEEE

investments. Second, large and complex applications can be executed on powerful servers instead of running stripped down versions on more resource constrained PDAs, providing greater functionality and better performance [10]. Third, applications with Internet connectivity requirements can take advantage of servers with faster networks and better connectivity, further boosting application performance. Fourth, PDAs can be even simpler devices since they do not need to perform complex application logic, potentially reducing energy consumption and extending battery life. Finally, PDA thin clients can be essentially stateless appliances that do not need to be backed up or restored, require almost no maintenance or upgrades, and do not store any sensitive data that can be lost or stolen. This model provides a viable avenue for medical organizations to comply with HIPAA regulations [6] while embracing mobile handhelds in their day to day operations.

II. PASSPORT Usage Model

PASSPORT consists of key components, a simple client viewer application that runs on the PDA and a server that runs on any commodity PC hardware. The server leverages more powerful PCs to to run all application logic. The client takes user input from the PDA stylus and virtual keyboard and sends them to the server to pass to the applications. Screen updates are then sent back from the server to the client for display to the user.

When the PASSPORT PDA client is started, the user is presented with a simple graphical interface where where information such as server address and port, user authentication information, and session settings can be provided. PASSPORT first attempts to connect to the server and perform the necessary handshaking. Once this process has been completed, PASSPORT presents the user with the most recent display of his application session. If the session does not exist, a new session is created. Existing sessions can be seamlessly continued without changes in the session setting or server configuration.

PASSPORT's persistent application session model enables a user to reconnect to a session from devices other than the one on which the session was originally initiated. This provides users with seamless mobility across different devices. If a user loses his PDA, he can easily use another PDA to access his application session. Furthermore, PASSPORT allows users to use non-PDA devices to access application sessions as well. A user can access the same persistent application session on a desktop PC as on a PDA, enabling a user to use the same session from any computer.

PASSPORT's persistent session model addresses a key problem encountered by mobile users, the lack of a common computing environment for running applications across computers. Computing environments often store important information such as personal settings, shortcuts, application specific settings to function in a much more useful manner. The problem that occurs when a user moves between computers is that this data cannot move with the user. Furthermore, applications often depend on other helper applications and libraries to execute, which may not be consistently available across all computers. PASSPORT addresses this problem by enabling a user to remotely use the exact same computing environment and applications from any computer. As a result, PASSPORT can provide a common, consistent computing environment for mobile users across different devices without requiring them to attempt to repeatedly synchronize different computing environments across multiple machines.

To enable a user to access the same application session on different devices, PASSPORT provides mechanisms to support different display sizes and resolutions. Toward this end, PASSPORT provides a zoom feature that enables a user to zoom in and out of a display and allows the display of an application session to be resized to fit the screen of the device being used. PASSPORT also provides mechanisms to support different display orientations. To accommodate PDA user preferences, PASSPORT provides an orientation feature that enables it to seamlessly rotate the display between landscape and portrait mode.

Because screen space is a relatively scarce resource on PDAs, PASSPORT runs in fullscreen mode to maximize the screen area available to display the application session. To be able to use all of the screen on the PDA and still allow the user to control and interact with the PDA, PASSPORT reuses the typical shortcut buttons found on PDAs to perform all the control functions available to the user.

III. PASSPORT System Architecture

PASSPORT builds on the pTHINC [9] remote display architecture to provide service providers a mechanism for streaming application display content from servers to PDAs. PASSPORT virtualizes the display at the server by leveraging the video device abstraction layer, which sits below the window server and above the framebuffer. This is a well-defined, low-level, device-dependent layer that exposes the video hardware to the display system. PASSPORT accomplishes this through a simple virtual display driver that intercepts drawing commands, and packetizes and sends them over the network to a client.

While other remote display approaches intercept display commands at other layers of the display subsystem, PASSPORT's display virtualization approach provides some key benefits in efficiently supporting PDA clients. PASSPORT's approach of intercepting at the device driver provides an effective balance between client and server simplicity, and the ability to efficiently encode and decode screen updates.

By using a low-level virtual display approach, PASSPORT can efficiently encode application display commands using only a small set of low-level commands. In a PDA environment, this set of commands provides a crucial component in maintaining the simplicity of the client in the resource-constrained PDA environment. The display commands are invoked as to mitigate the impact on network bandwidth.

To enable users to just as easily access their application sessions from a desktop computer at home as from a PDA while on the road, PASSPORT provides a resize mechanism to zoom in and out of the display of a session. PASSPORT resizing is completely supported by the server, not the client. The server resamples updates to fit within the PDAs viewport before they are transmitted over the network.

To enable users to orient their displays on a PDA for the best viewing experience, PASSPORT provides a display rotation mechanism to switch between landscape and portrait viewing modes. PASSPORT display rotation is completely supported by the client, not the server. PASSPORT does not explicitly recalculate the geometry of display updates to perform rotation, which would be computationally expensive. Instead, PASSPORT simply changes the way data is copied into the framebuffer to switch between display modes.

IV. Experimental Results

We have implemented a PASSPORT client and server prototype that supports widely-used Windows Mobile-based Pocket PC devices as clients and both Windows and Linux machines as servers. Our prototype enables service providers to deliver both Linux and Windows application services to handheld devices. We discuss two common application scenarios on PDAs to illustrate the differences between using PASSPORT versus running native PDA applications. The applications are web browsing and financial management. We present web browsing using PASSPORT with a Linux server and financial management using PASSPORT with a Windows server to demonstrate the flexibility that PASSPORT provides in delivering both Linux and Windows applications. All of the applications were accessed using a Dell Axim X51v PDA.

Figures 1 and 2 show screenshots of web browsing using PASSPORT and a full-function Mozilla Firefox web browser versus running Pocket IE natively on the PDA, respectively. Because of the limited resolution of the screenshots, they effectively show the layout differences between different platforms but do not reproduce the actual display quality of the PDA, which is much better than what is shown in these figures. Both screenshots show the same web page from BBC News, but display them very differently. PASSPORT provides the user with a wide range of display options, enabling the user to see the entire web page as well as zoom in on different parts of the web page. The result is a quality display experience similar to the familiar experience of web browsing on a desktop computer. PASSPORT enables the user to use a full-function desktop web browser on the PDA, providing robust support for viewing the same wide range of web sites that are accessible on a desktop computer. Furthermore, a quantitative comparison of web browsing performance shows that PASSPORT can be more than seven times faster than using a native PDA browser [9].

In contrast, running the native PDA application provides the user with a limited viewing experience of only being able to see a small portion of the web page at a time and needing to scroll around the web page frequently to view the content. Because the BBC News web page is not designed for viewing on PDAs, the native PDA web browser ends up only being able to display the top left corner of the web page when it is initially downloaded. This top left corner primarily consists of the BBC News logo, displaying very little useful content to the user. In addition, the Pocket IE user interface consumes a substantial amount of screen area, particularly the top and bottom menu bars, further reducing the available screen area for displaying useful web content. A bigger problem is that Pocket IE does not correctly parse parts of the BBC News web page depending on the particular web content being displayed. In scrolling around the BBC News web page shown, parts of the page are missing or misaligned. These problems are due to the resource restrictions of the PDA, resulting in the Pocket IE web browser having more limited functionality as a stripped down version of the equivalent Microsoft IE web browser that runs on a desktop computer. A wide range of web sites such as the BBC News web site cannot be displayed properly on the PDA using Pocket IE because of its incomplete support for commonly used web technologies such as Javascript.

Figures 3 and 4 show screenshots of running Quicken financial management software using PASSPORT and the full-function desktop version versus running Pocket Quicken natively on the PDA, respectively. PASSPORT provides the user with a quality display experience similar to the familiar experience of using Quicken on a desktop computer. PASSPORT enables the user to use the full-function desktop Quicken software on the PDA. As a result, users can access their Quicken data via PASSPORT across handheld devices and desktop computers without any need to maintain and attempt to synchronize multiple copies of their data across different devices.

In contrast, running the native PDA Pocket Quicken application provides the user with access to a very limited application compared to the original desktop version. Pocket Quicken is not capable of displaying in-depth financial analysis reports due to display resolution limitations and sub-par processing capabilities. Pocket Quicken is limited to maintaining short lists of expenses and viewing balances. Because of its limited functionality, Pocket Quicken also requires the desktop version to be installed on another desktop machine and needs to synchronize its data with the desktop version, requiring the user to purchase two versions of the software to provide financial management functionality on the PDA.

V. Related Work

Many thin clients have been developed and some have PDA clients, including Microsoft's Remote Desktop [3], Citrix MetraFrame XP [2], Virtual Network Computing [13], [12], GoToMyPC [5], and Tarantella [14]. However, these systems were first designed for desktop computing and retrofitted for PDAs as an afterthought. Unlike PASSPORT, they do not address key system architecture and usability issues important for PDAs. This limits their display quality, system performance, available screen space, and overall usability on PDAs. PASSPORT overcomes these limitations by building on recent work by two of the authors on THINC [1] and our recent work on pTHINC [9].

VI. Conclusions

PASSPORT provides a foundation which service providers can build upon to deliver application services on mobile handheld devices. PASSPORT's thin-client architecture provides key architectural and usability mechanisms such as server-side screen resizing, client-side screen rotation using simple copy

Fig. 1. **PASSPORT Web Screenshot: BBC News**

Fig. 2. **Native IE Screenshot: BBC News**

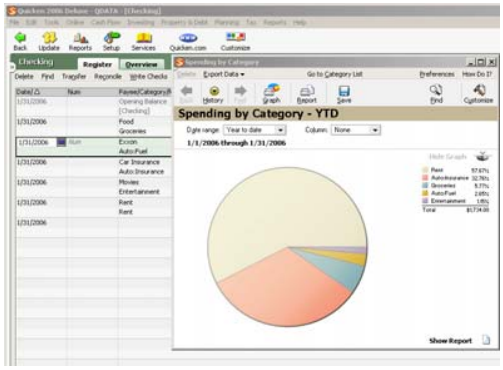

Fig. 3. **PASSPORT Application Screenshot: Quicken**

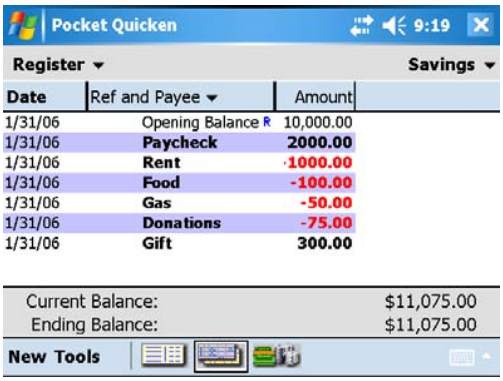

Fig. 4. **Native Application Screenshot: Pocket Quicken**

techniques, and maximizing screen space for display updates and leveraging existing PDA control buttons for UI elements. PASSPORT transparently supports streaming traditional desktop applications to PDA devices, providing mobile users with ubiquitous access to a consistent, personalized, and full-featured application services across heterogeneous devices. We have implemented PASSPORT to support both Windows and Linux applications without any application modifications. Our experiences with the system demonstrate that PASSPORT can provide a superior approach for delivery application services on mobile handheld devices.

VII. Acknowledgments

This work was supported in part by NSF ITR grants CCR-0219943 and CNS-0426623, Sun Microsystems, and an IBM SUR Award.

References

[1] R. Baratto, L. Kim, and J. Nieh. THINC: A Virtual Display Architecture for Thin-Client Computing. In *Proceedings of the 20^{th} ACM Symposium on Operating Systems Principles (SOSP)*, Oct. 2005.

[2] Citrix Metaframe. `http://www.citrix.com`.

[3] B. C. Cumberland, G. Carius, and A. Muir. *Microsoft Windows NT Server 4.0, Terminal Server Edition: Technical Reference*. Microsoft Press, Redmond, WA, 1999.

[4] A. Fox, I. Goldberg, S. D. Gribble, and D. C. Lee. Experience With Top Gun Wingman: A Proxy-Based Graphical Web Browser for the 3Com PalmPilot. In *Proceedings of Middleware '98, Lake District, England, September 1998*, 1998.

[5] GoToMyPC. `http://www.gotomypc.com/`.

[6] Health Insurance Portability and Accountability Act. `http://www.hhs.gov/ocr/hipaa/`.

[7] A. Joshi. On proxy agents, mobility, and web access. *Mobile Networks and Applications*, 5(4):233–241, 2000.

[8] J. Kangasharju, Y. G. Kwon, and A. Ortega. Design and Implementation of a Soft Caching Proxy. *Computer Networks and ISDN Systems*, 30(22–23):2113–2121, 1998.

[9] J. Kim, R. Baratto, and J. Nieh. pTHINC: A Thin-Client Architecture for Mobile Wireless Web. In *Proceedings of the 15^{th} International World Wide Web Conference (WWW 2006)*, May 2006.

[10] A. Lai, J. Nieh, B. Bohra, V. Nandikonda, A. P. Surana, and S. Varshneya. Improving Web Browsing on Wireless PDAs Using Thin-Client Computing. In *Proceedings of the 13th International World Wide Web Conference (WWW)*, May 2004.

[11] A. Maheshwari, A. Sharma, K. Ramamritham, and P. Shenoy. TranSquid: Transcoding and caching proxy for heterogenous ecommerce environments. In *Proceedings of the 12th IEEE Workshop on Research Issues in Data Engineering (RIDE '02)*, Feb. 2002.

[12] .NET VNC Viewer for PocketPC. `http://dotnetvnc.sourceforge.net/`.

[13] T. Richardson, Q. Stafford-Fraser, K. R. Wood, and A. Hopper. Virtual Network Computing. *IEEE Internet Computing*, 2(1), Jan./Feb. 1998.

[14] Sun Secure Global Desktop. `http://www.sun.com/software/products/sgd/`.