Extending Desktop Applications with Pocket-size Devices

Roberto Silveira Silva Filho
Institute for Software Research
University of California, Irvine
Irvine, CA 92697-3425
+1 (949) 824-4121
rsilvafi@ics.uci.edu

David F. Redmiles
Institute for Software Research
University of California, Irvine
Irvine, CA 92697-3425
+1 (949) 824-3823
redmiles@ics.uci.edu

ABSTRACT

Impromptu is a test bed for our usable security studies. It implements a peer-to-peer file sharing application that combines configuration, action and visualizations to help users make informed security choices. Previous user studies with Impromptu [1] indicated limitations in the tool. As different application windows tend to stay on top of Impromptu during a regular file sharing activity, the visualization of events in this interface gets, jeopardized. In this work, we discuss the use of PDAs as alternative information channels that complement the main desktop application. We describe our prototype, presenting our design rationale and the trade-offs in supporting awareness in a limited computer device.

1. INTRODUCTION

Our research has focused on the design of systems to support usable security. By relying on event-based monitoring, visualizations, and the integration of configuration and action in the development of applications, our goal is to create a technical infrastructure which makes visible the configuration, activity, and implications of available security mechanisms, thereby allowing end users to make informed security choices resulting in increased effective security. Those principles have been used in the design and implementation of a peer-to-peer file sharing application called Impromptu [2], that works as a test bed where different visualizations have been devised.

Our previous studies with Impromptu [1] indicated a need for extra peripheral information channels that could allow users to continuously visualize the activities of other peers. File sharing is not a means in itself, and relies on auxiliary tools such as word processors, spreadsheets, etc. When using those tools with Impromptu, users frequently maximize those applications, obliterating the Impromptu interface.

Whereas there are different ways of supporting peripheral awareness in applications, we are particularly interested the role of pocket-size PCs, or Personal Digital Assistants (PDAs for short) in augmenting desktop applications functionality, more specific as channels for awareness. The key insight is that those down-sized computers can provide extra visualizations or interfaces that operate in concert with their desktop counterparts, allowing users to have a peripheral view of the desktop application. In this work, we describe the architecture and interface of one of those thin clients interfaces.

2. IMPROMPTU DESKTOP APPLICATION

The Impromptu prototype [2] is a peer-to-peer file sharing tool designed to support co-located and ad-hoc meetings of small groups (typically no more than 10 people), where users participate mainly by sharing files. Files are shared according to different visibility levels expressed as concentric circles in a virtual table (see Figure 1). Impromptu integrates configuration and action concepts with activity visualizations that allow users to see when peers join and leave a session, and when they read or write to the user's files.

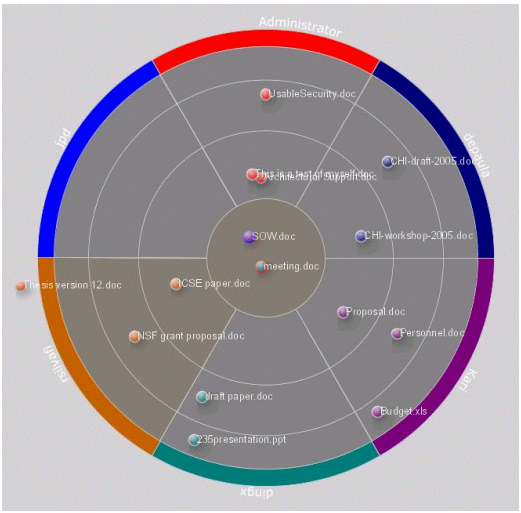

Figure 1: Impromptu client user interface

As seen on Figure 1, Impromptu provides a shared workspace in the form of a "pie" in which each "slice" corresponds to a single user's area of the workspace. These areas expand and contract as users arrive and leave a session. Files, represented by labeled dots, are placed in and around the circular region. Each area is tagged, on the pie's perimeter, with a unique color assigned for each user. This color is also associated with a user's files, and with indicators of that user's activity.

The pie in turn is separated into multiple concentric regions; the basic metaphor is that the closer the files are to the center, the "more shared" they are. Various degrees of sharing might be implemented. In this configuration, files outside the circle are not shared at all, but available to the local user only; files in the outer region are visible but not readable or writable to others; files in the next region are readable but not writable; in the next, readable and writable; and in the center, readable, writable, and available persistently. Persistent access means that the file remains accessible even after the owner leaves the session.

3. PDA TECHNICAL LIMITATIONS

In the design and implementation of our prototype (a.k.a. thin client), many factors impacted in the capabilities of the application and in the design of the interface. Those main factors include:

Processing and memory limitations. PDAs are typically limited in memory and processing. The target platform used in our experiments was a Dell PocketPC running Windows CE. It has 64MB RAM and 128MB of flash memory for binaries and data. The processor runs on a 350MHz clock.

Limited input and screen size. The input is limited in terms of text input, which is typically performed through a digital keyboard; and the screen resolution, which is QVGA (Quarter VGA), with a useful area of 229x225 pixels when the 81x 240 screen keyboard is not activated.

Programming language and runtime constraints. Since Impromptu was implemented in Java and RMI, the thin client was also implemented in Java. The JavaME-PP (Java Micro Edition Personal Profile) was selected. It is the most complete Java profile for such devices, and provides a limited subset of J2SE1.3.

As a consequence, in our design, we focused on presenting relevant events form the desktop application; while direct manipulation of artifacts (reading, writing and manipulating files), available in the desktop tool, were not supported. The GUI was implemented using more conventional AWT widgets. The result is a visualization pad where events can be viewed.

4. THIN CLIENT ARCHITECTURE

The thin client architecture (see Figure 2) relies on the same event-based design as the desktop tool. It uses YANCEES, the publish/subscribe infrastructure, that collects and routes events from the virtual group file repository. By listening to events generated by this application, the thin client builds its own view of the distributed system, and keeps itself consistent with the desktop application.

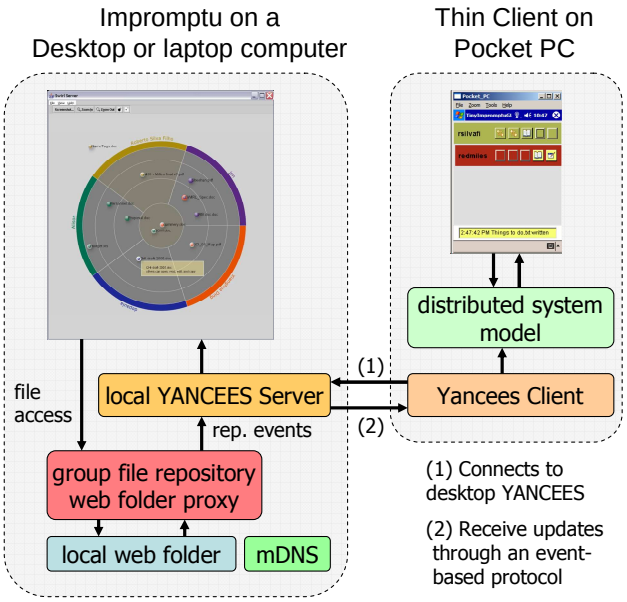

Figure 2 The Thin Client Architecture

5. THIN CLIENT GUI DESCRIPTION

In the design of the thin client interface, we adopted a ticker tape metaphor (see Figure 3), where icons representing events in the Impromptu application are displayed according to a temporal line (from right to left), as they get produced. By clicking on the event icon, the user can get more detailed information as displayed in the bottom of the screen. Whenever a new event is detected, its description is placed in the bottom of the screen, a sound is produced and the screen blinks, calling the attention of the user to the pocket PC, where the event can be further examined by clicking on its icon representation.

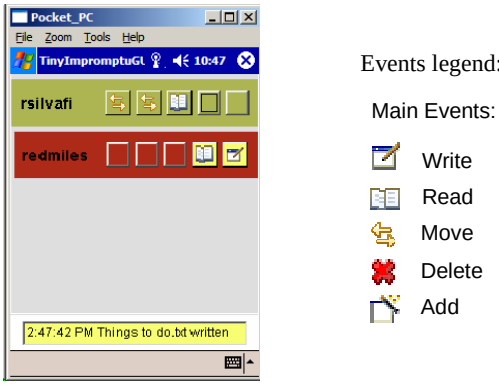

Figure 3 Ticker Tape Alternative Pocket PC Visualization

This interface allows the easier visualization of the desktop application events, which can drive the attention to the users back to the desktop tool, where direct file manipulation can be done. The current prototype, however is limited in the number of events it presents at once, five; and on the number of peers, five.

6. CONCLUSIONS AND RELATED WORK

Whereas the study of multi-modal interfaces and the integration of different devices has been studied before, for example, Meyers et. al. discusses the use of different devices in augmenting the regular desktop applications [3], our focus in this work is on the use of such devices to augment desktop applications with extra displays and modalities. We plan on performing further user studies to assess the impact of such approach in the augmentation of activity awareness during regular use activities.

7. ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under awards 0541462, 0133749, 0205724, 0326105, 0527729, and 0524033, and a grant from Intel Corporation.

8. REFERENCES

[1] R. DePaula, X. Ding, P. Dourish, K. Nies, B. Pillet, D. Redmiles, J. Ren, J. Rode, and R. S. Silva Filho, "Two Experiences Designing for Effective Security," presented at SOUPS 2005, Pittsburgh, PA, 2005.

[2] R. DePaula, X. Ding, P. Dourish, K. Nies, B. Pillet, D. Redmiles, J. Ren, J. Rode, and R. S. Silva Filho, "In the Eye of the Beholder: A Visualization-based Approach to Information System Security," *IJCHS - Special Issue on HCI Research in Privacy and Security*, vol. 63, pp. 5-24, 2005.

[3] B. Myers, R. Malkin, M. Bett, A. Waibel, B. Bostwick, R. C. Miller, J. Yang, M. Denecke, E. Seemann, J. Zhu, C. H. Peck, D. Kong, J. Nichols, and B. Scherlis, "Flexi-modal and multi-machine user interfaces," presented at IEEE Fourth International Conference on Multimodal Interfaces, Pittsburgh, PA, 2002.