

Individual Use of Hand-Held and Stationary Computers Simultaneously

Brad A. Myers, Robert C. Miller, Carl Evankovich, and Benjamin Bostwick

Human Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
bam@cs.cmu.edu
<http://www.cs.cmu.edu/~pebbles>

ABSTRACT

Increasingly, people will be in situations where there are multiple communicating computing devices that have input / output capabilities. We have developed a suite of applications to study user interfaces that span both hand-held computers such as Palm Pilots and regular computers such as a PC. Previous work on handhelds has reported on applications that support group work. Here, we focus on how multiple devices can enhance the user interface for a single user. The hand-held's screen can be used as a customizable input and output device, to provide various controls for existing and new applications. For example, the hand-held can contain scroll bars, buttons, virtual knobs and menus. It can also display the slide notes or a list of slide titles for a presentation, the list of current tasks and windows, and lists of links for web browsing. The user can tap on these lists on the hand-held to control the PC.

Keywords: Personal Digital Assistants (PDAs), Hand-held computers, Palm Pilot, Windows CE, Two-Handed Input, Smart Environments, Ubiquitous Computing .

INTRODUCTION

The age of *ubiquitous computing* [21] is at hand with computing devices of different shapes and sizes appearing in offices, homes, classrooms, and in people's pockets. Many environments contain embedded computers and data projectors, including most offices, many classrooms (such as the Classroom 2000 [1]), meeting rooms, and even homes. These are often called "Smart Environments" or "Smart Houses" [11]. One little-studied aspect of these environments is how personal, hand-held computers will inter-operate with the stationary computers. More and more people are carrying around programmable computers in the form of Personal Digital Assistants (PDAs) such as

a Palm Pilot or Windows CE device, or even in their cell-phones and watches. We are researching the question of what can users do with their hand-held computers in such an environment.¹

Some researchers have looked at using hand-held devices in group settings to support collaborative work [1] [7] [16] [18] [20]. However, there has been little study of how portable devices can work in concert with stationary computers to help individuals. Most work on hand-held devices for individuals has instead focused on how they can be used to *replace* stationary computers, for when the stationary computer is not available. This paper discusses *individual* use of the hand-held and the stationary computer *together* when are both available, which has not been addressed by prior work.

There are many situations when the hand-held computer's display, input mechanisms, and processor can be used to enhance the stationary computer's application. For example, the hand-held can provide extra views on the data on the PC, and there can be buttons on the hand-held that control the PC's applications. This paper presents an overview of all the applications we have created for individual use of hand-helds and PCs together.

The next section provides an overview of our approach, followed by related work, and then our new applications are presented. Finally, we discuss general observations and our plans for future work.

OVERVIEW

In the "old days," computers had a variety of input switches and knobs. Today, computers are standardized with just a keyboard and a mouse for input, and connecting custom input devices is difficult and expensive. Although

Submitted for Publication

¹ This paper uses the terms PDAs, hand-held computers, and portable computers interchangeably, since most of the ideas in this paper would equally apply to any of these devices. Our research has so far focused on using PDAs such as Palm Pilots and Windows CE devices. We will use "PC" to refer to the stationary computer, which might actually be a laptop and therefore not necessarily stationary.



Figure 1. Typical configuration, with a PDA in its cradle on the left of the keyboard, and the user's hands on the PDA and the mouse.

today's computers have high-resolution screens and window managers, users can still need extra space to display information. For people who have already purchased a PDA, we want to exploit it to provide the benefits of having an extra input and output device. For example, Figure 1 shows the PDA being used at the same time as the mouse as an extra input and output device for the non-dominant hand which has been shown to be effective [15].

Most modern PDAs come with a cradle to synchronize the data with the PC (see Figure 1) so having a PDA communicate with a PC is easy. When using a cradle is inconvenient, a serial cable wire can be purchased. Some of the newest PDAs such as the Palm V and the HP Jornada 420 have rechargeable batteries which are recharged when the device is in its cradle. Therefore, the user is *supposed* to leave the device connected to the PC whenever the user is next to the PC. Many PDAs also have built-in wireless communication in the form of infrared (IR), which many laptops and some desktop computers support as well. Future wireless technologies such as BlueTooth [9] and HomeRF [17] will make connecting multiple devices even easier.

The PDA can be used to extend the desktop in various ways. It can serve as a customizable input device, with "soft" buttons, sliders, menus and other controls displayed on the screen. These can be made big enough to operate with a finger, even with the non-dominant hand (as in Figure 1), and these interfaces can then be carried with the user, and even used with different PCs. The PDA can also be used as an output and control device to provide secondary views. This is useful when the entire PC screen is engaged and unavailable, such as during a PowerPoint presentation where we display the notes of the current slide on the PDA. Another use is to display information that should not be covered by other windows, for example to display a list of tasks and windows on the PDA to support easy switching.

To demonstrate our ideas, we developed a wide variety of PDA applications, most of which have been made available for free on the World-Wide Web. All of these work with off-the-shelf PC programs. Some of these applications include:

- **Slideshow Commander:** displays the slide notes, titles and thumbnail images of the slides on the PDA and allows the user to control the show from the PDA.
- **Scrollers:** acts as an input device to scroll the current window on the PC. The scrollers on the PDA can even be used with the left hand, which research suggests can be very effective [22].
- **Switcher:** displays a list on the PDA of the current tasks (like the Windows Taskbar) and a list of the windows in each task (like the Windows menu item in some applications). This eliminates the confusion of which of these very different mechanisms to use to get to a desired window, and provides a consistent, quick, and always-available mechanism. The user can tap on an item on the PDA to cause that window to come to the front of the PC.
- **Remote Clipboard:** supports cutting and pasting information between the PC and the PDA, either as full data or as links, to make data exchange easier [13]. This application also supports opening files or URLs in their appropriate application. This can be used to make a list on the PDA of common applications to invoke, similar to the "Quick Launch" in Windows 98.
- **Web Assistant:** stores links copied from the current web page to facilitate the "hub and spoke" style of browsing, which is quite common [5].
- **Shortcutter:** allows the user to design screens of keyboard shortcuts, macros, and strings on the PDA, and use these to control any PC application. The buttons can be big enough to hit with a finger, or tiny so that many will fit on a screen. The Shortcutter can provide customizable interfaces on the PDA even for applications that do not have a customization facility on the PC. Since these are on the PDA, you can take them with you and use them even on other people's computers.

RELATED WORK

The Xerox ParcTab [20] project investigated using small hand-held devices at the same time as other computers. The ParcTab was used to investigate some aspects of remote cursors and informal voting about how well the speaker was doing. Bill Buxton conceived of using the ParcTabs as an "orb" in the left hand to control scrolling and page turning, while using a stylus on the large LiveBoard with the right hand,² but apparently this application was never implemented. Our scrollers and Shortcutter are the first implementation of this idea, and are flexible enough for investigating many such scenarios of use.

The M-Pad system [18] supports multiple users collaborating with PalmPilots and a large whiteboard. It allows

² The analogy is that in formal pictures, the Queen holds a scepter in her right hand and a ball called an "Orb" in her left.

information to be transferred back and forth between different devices using “pick and drop,” which is related to our support for cutting and pasting information between the PC and the PDA. M-Pad requires an extra position sensor in addition to a PalmPilot and was designed for use by multiple people.

There have been many investigations of using both hands at the same time to control a computer. For example, an early study [4] showed that people could effectively scroll and change objects’ size with the left hand while positioning with the right hand, and that many people operated both devices at the same time. That study used custom hardware as the scrollers. A recent IBM study [22] found that two techniques were about the same speed in a browsing task: a one-handed technique for scrolling using a pointing-stick on a mouse, and two-handed scrolling. Another study investigated how accurately gestures could be drawn with the non-dominant hand on a small touchpad mounted on top of a mouse [2]. A special gesture pad in Shortcutter is designed to allow real actions to be associated with these kinds of gestures. A new study investigates performance parameters of using Shortcutter to create various scrollers on the PDA [15].

Cross-application macros, such as provided by Shortcutter have been available in other applications, such as the Macintosh programs QuicKeys from CE Software or Tempo from Affinity Microsystems. The Tatlin system [12] investigates how to use AppleScript to create macros across applications by example. However, Shortcutter adds several new kinds of actions to the script, and the ability to keep the scripts on a PDA which can be easily transported.

Other groups are studying the use of PalmPilots in various settings, where they are *not* connected to a PC. For example, NotePals synchronizes people’s independent notes after a meeting [7], and Georgia Tech’s “Classroom 2000” project [1] is studying the use of hand-held computers in classrooms. For PDAs connected to a PC, a previous paper [16] discusses the use of multiple PDAs connected to a PC to support meetings. For example, each PDA can be used as the mouse and keyboard of the PC, so users can take turns entering data or controlling the PC. The current work shows that a PDA can be equally useful for a single person as an extension to desktop applications, and discusses the various ways we have extended the PC desktop user interface using the PDA.

EXAMPLE APPLICATIONS

We have created a wide range of applications to investigate how a PDA can extend desktop user interfaces. The following sections discuss some of the applications we have created.

Slideshow Commander

When giving presentations from a computer using slide show programs such as Microsoft’s PowerPoint, the

speaker will often use notes. These are usually printed on paper, because slide programs take over the full PC display, which prevents the screen from showing the notes.³ Furthermore, the speaker usually uses the mouse or keyboard to move forward and backward through the slides, which means that the speaker must be close to the PC. In a Contextual Inquiry study of PowerPoint presentations [6], other problems were identified, including the desire walk away from the presentation computer to be closer to the audience.

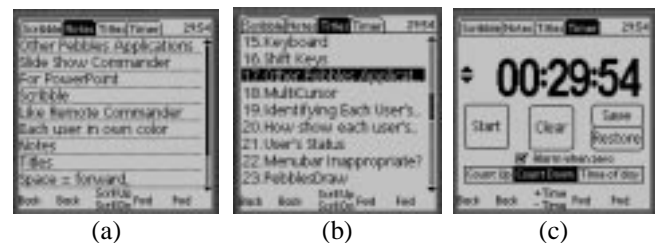


Figure 2. The “Notes,” “Titles,” and “Timer” panels of the Slideshow Commander on the PalmPilot.

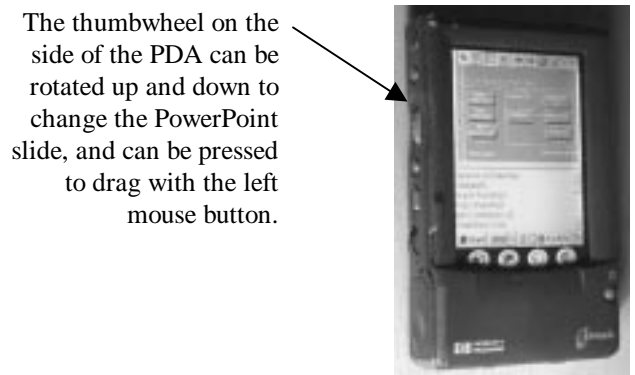


Figure 3. The Notes and Thumbnail view of the Windows CE version of Slideshow Commander running on a color HP Jornada 420.

The Slideshow Commander application solves some of these problems. It works with Microsoft PowerPoint 97 and either a PalmPilot or a palm-size Windows CE machine. The PDA can be connected to the PC using a long serial wire or by infrared, and in the near future, better wireless connections will be available. Slideshow Commander has various panels that support different tasks during a slide show.

One panel allows the user to point and scribble on the PC screen by writing on the PDA. Moving a finger or the stylus across the surface of the PDA moves the cursor on the PC the same relative amount, as with a touchpad. More information on how the PDA can be used as a keyboard and mouse appears elsewhere [16].

³ Some versions of PC slide show programs and some video cards will allow the projector to display the slides while the PC’s display shows the notes, but this is not commonly available or used. Another option with PowerPoint is to use two different PCs connected by a serial cable, but this is quite awkward to set up.

The notes view (Figure 2a and 3) displays the notes associated with the current slide and optionally a thumbnail picture of the slide as well. The notes page is re-filled whenever the slide is changed, so it always displays the notes for the current slide. PowerPoint allows each slide to have associated notes that some people use—especially when the slides consist mostly of pictures. Other people put most of their text on the slides as bullet points. The notes view supports both styles of presentations. The user can choose whether to display the text that appears on the slide itself, the text from the notes associated with that slide, or both. When both are displayed, a line separates them in the notes display. If the notes do not fit on one screen, the scroll bar is enabled. For the thumbnail picture to be legible on gray-scale PDAs, we have PowerPoint generate a “black-and-white” picture of the slide, which removes the background.

The titles view (see Figure 2b) displays the titles of all the slides in the current talk. The currently displayed slide is always highlighted, and tapping on a slide name changes the presentation to that slide. This might be useful for people who have a large slide set and want to dynamically choose which slides to use for a given talk. Another use is at the end of the talk, during questions, to jump back to a specific slide under discussion.

A final pane (see Figure 2c) displays a timer, which can count up or down or display the current time of day. This is useful for timing the talk.

In any of the views, the user can still change slides by using the physical buttons on the Palm or the scroll wheel on the Windows CE (see Figure 3), or by giving a Graffiti gesture.

Scrollers

Early research [4] showed that people could effectively scroll using their non-dominant hand. Recently, there has been a profusion of devices to help with scrolling, including the Microsoft IntelliMouse, the IBM ScrollPoint mouse, and the middle mouse button that can be programmed to perform scrolling. These mechanisms all use the dominant hand. The Pebbles scrolling applications allow the PDA to be used as a scrolling device in either hand. Figure 4 shows some of the scrollers we have created. New research [15] shows that people can scroll effectively with the PDA in their left hand.

Although these applications were designed to support scrolling, they can also be used as an extra, parallel input stream from the mouse. For example, in a positioning and resizing task [4], the scroll events can be mapped to change the object’s size, to support two-handed operation. In the future, we will be experimenting with other tasks that require more than two degrees of freedom, such as translation and rotation in 3-D worlds.

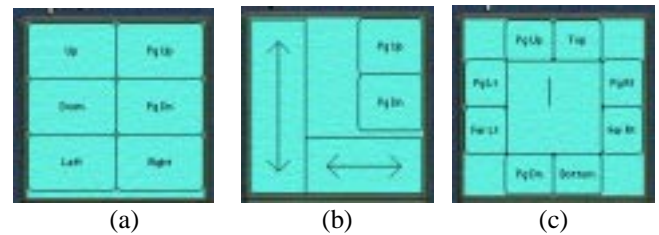


Figure 4. Three ways Pebbles provides for scrolling: (a) Using buttons that auto-repeat to scroll up and down a line or a page, or left and right. (b) Using a slider, where dragging a finger or stylus in a rectangle drags the text the same amount. (c) Using a virtual rate-controlled joystick, where pressing in the center and moving out scrolls the text in that direction at a rate proportional to the distance.

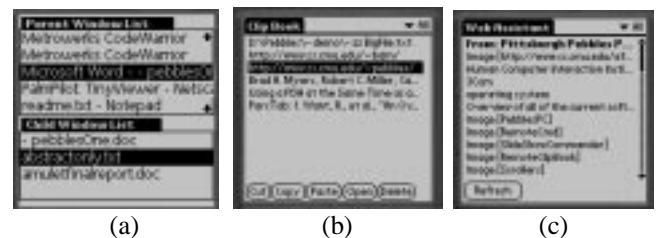


Figure 5. (a) Task switcher displays top-level tasks and then windows for the selected task. (b) Remote Clipboard connects the PC’s clipboard and the PDA’s. Here is a list of items pasted from the PC, which can be copied back to the PC or to another PDA application. The “Open” button causes the selected file or URL to be opened on the PC. (c) Web Assistant displays links on the current page. Clicking on a link causes the browser to go to that page. The list of links is only updated when the Refresh button is hit. For images which are links, the “alt” text is displayed in square brackets next to the word “Image” but if there is no “alt” text, then the URL is displayed instead.

Task and Window Switcher

Windows 95 and later versions provide a “Taskbar” at the bottom of the screen to switch among applications. Additionally, many applications have a “Windows” menu item to switch among windows in that application. The Switcher application (see Figure 5a) combines both capabilities into a single consistent user interface on the PDA. The advantage is that there is an easy and quick way to switch applications and windows, which can be operated with one or two clicks. Another advantage is that we can optionally group windows by application, even for multiple instances of the application. Since Switcher provides various ways of organizing the application’s windows, this can make it easier to find a window. For example, if there are multiple instances of Microsoft Word running, all the documents from all of them can be combined into one alphabetical list.

Remote Clipboard

PDA's generally have some sort of synchronization mechanism to keep the information on the PDA consistent with a PC. However, the process is somewhat inconvenient and time consuming. For example, on the Palm Pilot you need to HotSync, which copies all applications' data that have changed, and takes up to a minute.

To provide a quicker way to copy small amounts of data between the PDA and the PC, we developed Remote Clipboard [13], which connects together the clipboards of the PC and the PDA. Whenever the user copies or cuts text on either machine, it can be pasted on the other. File names or URLs can also be pasted onto the PDA, and the Open button on the PDA will cause the PC to open the application associated with the file, or open the web page in the default browser.

This allows information on the PC to be copied to the PDA either by value or by reference. If the user copies the information itself and pastes it to the PDA, this corresponds to passing the information "by value." If the user copies the filename or URL of the information, then it is passed "by reference." In this way, the PDA can work as an "Informative Thing" [3] and carry pointers to information that resides elsewhere.

The Remote Clipboard PDA application (see Figure 5b) provides one place to store the data on the PDA, but the information can be pasted and copied from any PDA applications, including the address book, scheduler, MemoPad, etc.

Web Assistant

Research [5] and experience show that web browsing often takes on a "hub and spoke" style, where the user frequently returns to a main index page in order to find the next out-link to click on. Examples include the results of a search, and table of contents and maps of web sites and on-line documents. The Pebbles Web Assistant application aids in these tasks by allowing links from the "hub" page to be copied to the PDA.

Figure 5c shows a view of the list of links on the PDA. Clicking on any link causes the web browser on the PC to switch to the specified page. Note that this does *not* automatically refill the PDA display with the links from the new page—it continues to show the original set of links so the user can easily move from link to link. Pressing the Refresh button refills the PDA page with the current page's links.

Many web pages are filled with irrelevant links. For example, a search results page can easily have more advertisement links than results. Therefore, the user can select a region of the browser text and only copy the links from that region onto the PDA. In the future, we expect to integrate more intelligent parsing technology [14] into the

Web Assistant so the useful links can be selected and copied more automatically.

Shortcutter

The Shortcutter application combines many of the features of the previous utilities to allow users to create custom panels of "shortcut" buttons, sliders, knobs, and pads. In edit mode, users can draw panels and assign an action to each item in the panel. Switching to run mode, the items will perform their actions.

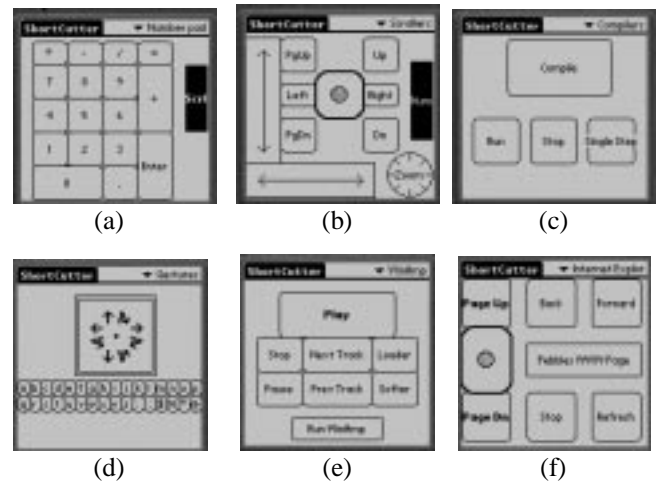


Figure 6. Panels created with Shortcutter. (a) A numeric keypad, (b) a collection of scrollers and a knob, (c) buttons for controlling any of a set of compilers, (d) a gesture pad and two rows of small buttons, (e) a controller for the WinAmp PC program, and (f) a panel for browsing in Internet Explorer.

We have defined several kinds of widgets that can be added to panels:

- **Buttons:** The user can assign a label, and various shapes and patterns. Buttons can be specified to auto-repeat if the user presses and holds on them, or to only execute once per tap.
- **Scrolling Widget:** The vertical and horizontal slide scrollers (see Figure 4b above) and the rate-control joystick scroller (see Figure 4c) are also available in the Shortcutter (see Figure 6b which includes two sliders, a rate-control joystick in the center, and some scroll buttons).
- **Mouse Pad:** This supplies a rectangular area where the user drags the stylus to perform mouse movements like a trackpad.
- **Gesture Pad:** This item allows users to give gestures such as those used in previous studies [2]. Currently, the supported gestures are hardwired, and are limited to simple strokes in various directions. Figure 6d shows the gesture pad supporting nine gestures: tap (dot), strokes in four directions, and back-and-forth strokes in four directions. Each of these gestures can be assigned a

different action. In the future, we might support more elaborate gestures or even user-trainable gestures.

- **Knobs:** For knob widgets, the user can specify one action to be generated when the user makes a clockwise gesture and another for counter-clockwise. These actions can be repeatedly sent as the gesture continues, to give the effect of “turning” the knob. For example, a keyboard key that increments a value might be assigned to counter-clockwise (such as the “+” key in Quicken, or a string like “+1=” for a calculator) and the opposite for the other direction. Since this is a general facility, any action can be assigned. Figure 6b shows a knob for zooming.

For many of these widgets, the user can choose any of the following actions:⁴

- **Key:** Send any PC keyboard key to the PC as if the user had typed it, including function and other special keys and modifiers such as ALT and Control. This makes it easy to create a panel like a numeric keypad (Figure 6a) which might be useful for laptops which do not have one. Anything that can be invoked using keyboard keys (menu items, etc.) can be easily assigned to a Shortcutter button.
- **String:** Send a string to the PC as if the user had typed it on the keyboard. This might be useful for creating buttons that serve as abbreviations, or as input to the PC during a Macro.
- **Open File/URL:** To tell the PC to open a file in its application, or to go to a specified page in a browser.
- **Run Application:** Causes a specified application to be run. The user can type the name of the .EXE file to be run, or else Shortcutter can be asked to try to determine the appropriate executable for the topmost application currently running on the PC.
- **Scrolling:** The scrolling commands for the scroll buttons (Figure 4a) use special Windows messages, and this action allows them to be assigned to items.
- **Switch Panel:** The user can create a button that goes to a particular panel. This does not send anything to the PC. The black buttons on the right of Figures 6a and 6b switch between these two panels.
- **Mouse Button:** This is for sending mouse button events, such as left or right buttons down and double-clicking. Parameters include the modifier keys if any (to support events like SHIFT-CLICK).
- **Recorded Event:** This uses a clever trick on the PC that records the next Windows message sent by a menu item or toolbar button. The button will then cause that action

to be executed at run time by sending the same message. This allows buttons on the PDA to perform actions that may not have a keyboard equivalent in the application, such as changing modes in a drawing program.

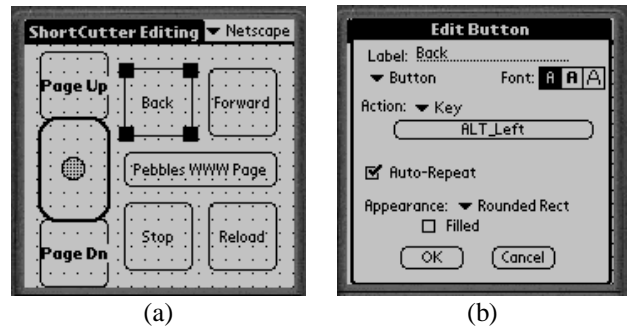


Figure 7. (a) The editing mode of Shortcutter with the Back button selected. (b) Setting the properties of the Back button so when you click on the Back button at run-time, it sends to the PC the keystroke Alt-left-arrow.

- **Wait for User:** This pops up a window with a button that waits for the user to tap OK. It is only really useful in Macros (see next item).
- **Macro:** This allows a sequence of the above actions to be associated with a button. Note that these macros can operate across multiple PC applications. Macros are edited on the PDA using conventional drag-and-drop and cut-and-paste techniques. Macros can invoke other macros, which supports procedural abstraction.
- **Application-Specific:** A button can have multiple actions associated with it, where each action is specific to a different PC application. Then, if the button is pressed, Shortcutter checks which PC application is on top, and chooses the appropriate action. This can provide a uniform, virtual interface to a set of applications that perform the same function but have disparate interfaces. An example use for this is that we use a variety of programming environments that unfortunately have different key assignments for actions and are not customizable. We created a panel of application-specific buttons that send the appropriate key for the current compiler (see Figure 6c).

The panels are constructed on the PDA by switching into “Edit Mode” (see Figure 7a). This supplies a conventional direct-manipulation editor, where buttons can be simply drawn, and their labels and properties assigned. The property sheet for a button is shown in Figure 7b. For the more complicated actions, other forms are used.

There are a number of special features of Shortcutter that make it even more useful:

- The hardware (physical) buttons on the PDA can be mapped to any action.
- Characters entered using Graffiti or the pop-up keyboard can be sent through to the PC (so the PDA can be used

⁴ Many of these actions use special PC mechanisms to achieve their effects, so they may not work in all situations, across different PC operating systems and applications.

as an extra keyboard), or mapped to any action described above.

- The background of each panel can be changed to make panels more attractive and distinguishable.
- Panels and items can be stored on the PC and reused, so particularly useful ones can be shared.
- Users can design buttons that are large enough to be hit with a finger (as in Figures 5 and 6), or very small so many items can fit on a panel and can be invoked using the stylus (as in the bottom of Figure 6d).

The result is a very flexible and useful application with which users can create many interesting personalized shortcuts that should make their everyday use of a PC more effective.

STATUS

All of the Pebbles applications discussed here are available for downloading off of the World-Wide Web at <http://www.cs.cmu.edu/~pebbles>. The Slideshow Commander and scrollers have been available since December, 1998, and the other applications were released just recently. The various Pebbles applications have been downloaded a total of about 15,000 times in the last year and a half.

Experience with the Slideshow Commander has been very positive, and many people are using it to give presentations. The Windows CE version of the Slideshow Commander program won first place in the utilities category of the 1999 ACM Quest for Windows CE Applications contest.

Third parties are also picking up on the Pebbles architecture. A commercial company, Iron Creek Software [10], is using the Pebbles architecture to build a new Palm interface to the popular WinAmp PC program for playing MP3 and other digital music. WinAmp can play in the background while user is doing other work on the PC, or WinAmp has full screen animations based on the music. In either case, it is useful to have a controller that uses a different screen, as supported by the Pebbles architecture. Figure 6e shows a Shortcutter WinAmp controller, but the Iron Creek version will also support downloading and rearranging play-lists on the PDA.

GENERAL OBSERVATIONS AND EXPERIENCE

The situations in which a combined PDA-PC user interface make the most sense are either when the user is using the PC from a distance, or the PC's screen is busy with other tasks. The Slide Show Commander is a good example of fitting both criteria. We have heard from users that often they want to control WinAmp from a distance, for example when across the room from the PC, so this is another good fit. People report that they want to read their

email and browse the web while leaning back in their chairs, so PDA applications that support this are useful.

Our applications like Task Switcher enable users to fill the PC's screen with the main application, and eliminate the space for the Taskbar.

Some of our applications get around limitations or problems with current versions of the PC programs. For example, all compilers should use the same keyboard shortcuts, eliminating the need for the panel in Figure 6c. However, it is likely that these kinds of problems will always arise, so even if some problems are fixed, others are likely to arise.

Another issue is how the PDA and the PC applications can be improved so they work better together. It would be useful if more PC applications provided a procedural interface in the same was a PowerPoint 97, so that they can be queried and controlled remotely. The main problem with the PDA is its tiny screen, so, for example, some people find that not enough of the notes of the Slide Show can fit on one screen-full. Also, the digitizer is not very sensitive to the pad of the finger, so it is best to use a fingernail, which is awkward for some people.

FUTURE WORK

Current work in the Pebbles project is focusing on two areas: mixing private and public information and classroom use. In many public meetings, it is useful for individuals to privately on their hand-held get more details about publicly displayed information on a wall display. Alternatively, the user might have additional details on their hand-held that they want to combine with the public record. Initial investigations in this area have been performed by others [8], but much more work is needed.

The other area of work will be on transitioning some of these applications to the classroom. The Slideshow Commander might broadcast to all the audience's computers the thumbnail picture and the notes of the current slide, to facilitate notetaking and understanding. A Contextual Inquiry analysis shows many places where this would be useful [6]. Another classroom application is support for testing on the students' hand-helds with immediate grading by the teacher's PC. The ubiquitous wireless network at our university will make these classroom applications feasible now, while highlighting interesting networking and security issues that have not been addressed by other related projects [19].

We will also be working on communicating to the PDA using RF wireless. It would be possible to port the PC side to also work on Macintosh and Unix computers.

CONCLUSIONS

The Pebbles project is exploring the many ways that PDAs can be used at the same time as stationary computers to

enhance the user's effectiveness. Surprisingly, this is an area that has received very little study. Perhaps this is because earlier PDAs drained their batteries quite quickly when communicating with the PC. With the new models that recharge in their cradles, this problem is eliminated, and in fact, it becomes the recommended practice to keep the PDA connected to the PC. Therefore, applications such as those described here become more compelling. We have found that the IR capabilities built into today's PDAs do not support these kinds of applications well, because IR needs to be carefully aimed, and most of these applications require the PDA to be at a comfortable position parallel to the PC, rather than pointing towards its IR port. However, low power radio frequency wireless communication for small devices is coming soon, with technologies like Bluetooth and HomeRF. When these become commonplace, the use of multiple devices at the same time, while they are all communicating, will become increasingly common. The applications described here point out a number of ways that these devices can be usefully integrated.

ACKNOWLEDGMENTS

For help with this paper, we would like to thank [Andrew Faulring](#), [Brad Vander Zanden](#), [Karen Cross](#), [Rich McDaniel](#), and [Bernita Myers](#).

The Pebbles project is supported by grants from DARPA, Microsoft, IBM and 3Com. This research was performed in part in connection with Contract number DAAD17-99-C-0061 with the U.S. Army Research Laboratory. The views and conclusions contained in this document are those of the authors and should not be interpreted as presenting the official policies or position, either expressed or implied, of the U.S. Army Research Laboratory or the U.S. Government unless so designated by other authorized documents. Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

REFERENCES

1. Abowd, G.D., *et al.* "Investigating the capture, integration and access problem of ubiquitous computing in an educational setting," in *Proceedings SIGCHI'98: Human Factors in Computing Systems*. 1998. Los Angeles, CA: pp. 440-447.
2. Balakrishnan, R. and Patel, P. "The PadMouse: Facilitating Selection and Spatial Positioning for the Non-Dominant Hand," in *Proceedings SIGCHI'98: Human Factors in Computing Systems*. 1998. Los Angeles, CA: pp. 9-16.
3. Barrett, R. and Maglio, P.P. "Informative Things: How to Attach Information to the Real World," in *Proceedings UIST'98: ACM SIGGRAPH Symposium on User Interface Software and Technology*. 1998. San Francisco, CA: pp. 81-88.
4. Buxton, W. and Myers, B. "A Study in Two-Handed Input," in *Proceedings SIGCHI'86: Human Factors in Computing Systems*. 1986. Boston, MA: pp. 321-326.
5. Card, S.K., Robertson, G.G., and York, W. "The WebBook and the Web-Forager: An Information Workspace for the World-Wide Web," in *Proceedings CHI'96: Human Factors in Computing Systems*. 1996. Vancouver, BC, Canada: pp. 111-117.
6. Cross, K., Warmack, A., and Myers, B. "Contextual Inquiry Analysis of PDA Presentation Software," in 1999. pp. Submitted for Publication.
7. Davis, R.C., *et al.* "NotePals: Lightweight Note Sharing by the Group, for the Group," in *Proceedings, CHI'99: Human Factors in Computing Systems*. 1999. Pittsburgh, PA: ACM. pp. 338-345.
8. Greenberg, S., Boyle, M., and Laberg, J., "PDAs and Shared Public Displays: Making Personal Information Public, and Public Information Personal." *Personal Technologies*, 1999. pp. To appear. March.
9. Haartsen, J., *et al.*, "Bluetooth: Vision, Goals, and Architecture." *ACM Mobile Computing and Communications Review*, 1998. 2(4) pp. 38-45. Oct. www.bluetooth.com.
10. IronCreek, "PalmAmp by IronCreek Software, palmamp@ironcreek.net," 1999. Palm interface to WinAmp: www.winamp.com.
11. Levy, S., "The New Digital Galaxy; A Really Smart House." *Newsweek*, 1999. (22)
12. Lieberman, H. "Integrating User Interface Agents with Conventional Applications," in *1998 International Conference On Intelligent User Interfaces*. 1998. San Francisco, CA: pp. 39-46.
13. Miller, R.C. and Myers, B. "Synchronizing Clipboards of Multiple Computers," in *Proceedings UIST'99: ACM SIGGRAPH Symposium on User Interface Software and Technology*. 1999. Asheville, NC: pp. To appear.
14. Miller, R.C. and Myers, B.A. "Lightweight Structured Text Processing," in *Usenix Annual Technical Conference*. 1999. Monterey, California: pp. 131-144.
15. Myers, B.A., Lie, K.P.L., and Yang, B.-C.J. "Two-Handed Input Using a PDA And a Mouse," in 1999. pp. Submitted for Publication.
16. Myers, B.A., Stiel, H., and Gargiulo, R. "Collaboration Using Multiple PDAs Connected to a PC," in *Proceedings CSCW'98: ACM Conference on Computer-Supported Cooperative Work*. 1998. Seattle, WA: pp. 285-294.
17. Negus, K.J., *et al.*, "HomeRF and SWAP: Wireless Networking for the Connected Home." *ACM Mobile Computing and Communications Review*, 1998. 2(4) pp. 28-37. Oct. www.homerf.org.
18. Rekimoto, J. "A Multiple Device Approach for Supporting Whiteboard-based Interactions," in *Proceedings SIGCHI'98: Human Factors in Computing Systems*. 1998. Los Angeles, CA: pp. 344-351.
19. Truong, K.N. and Abowd, G.D. "StuPad: Integrating Student Notes with Class Lectures," in *CHI'99 Extended Abstracts: Human Factors in Computer Systems, Extended Abstracts*. 1999. Pittsburgh, PA: pp. 208-209.
20. Want, R., *et al.*, "An Overview of the ParcTab Ubiquitous Computing Experiment." *IEEE Personal Communications*, 1995. pp. 28-43. December. Also appears as Xerox PARC Technical Report CSL-95-1, March, 1995.
21. Weiser, M., "Some Computer Science Issues in Ubiquitous Computing." *CACM*, 1993. 36(7) pp. 74-83. July.
22. Zhai, S., Smith, B.A., and Selker, T. "Improving Browsing Performance: A Study of Four Input Devices for Scrolling and Pointing," in *Proceedings of Interact'97: The Sixth IFIP Conference on Human-Computer Interaction*. 1997. Sydney, Australia: pp. 286-292.