

# Linking and Messaging from Real Paper in the Paper PDA

Jeremy M. Heiner, Scott E. Hudson, Kenichiro Tanaka

Human Computer Interaction Institute

Carnegie Mellon University

Pittsburgh, PA 15213

{heiner, hudson}@cs.cmu.edu, kt12@andrew.cmu.edu

## ABSTRACT

It is well known that paper is a very fluid, natural, and easy to use medium for manipulating some kinds of information. It is familiar, portable, flexible, inexpensive, and offers good readability properties. Paper also has well known limitations when compared with electronic media. Work in hybrid paper electronic interfaces seeks to bring electronic capabilities to real paper in order to obtain the best properties of each. This paper describes a hybrid paper electronic system – the *Paper PDA* – which is designed to allow electronic capabilities to be employed within a conventional paper notebook, calendar, or organizer. The Paper PDA is based on a simple observation: a paper notebook can be synchronized with a body of electronic information much like an electronic PDA can be synchronized with information hosted on a personal computer. This can be accomplished by scanning, recognizing and processing its contents, then printing a new copy. This paper introduces the Paper PDA concept and considers interaction techniques and applications designed to work within the Paper PDA. The *StickerLink* technique supports on-paper hyperlinking using removable paper stickers. Two applications are also considered which look at aspects of electronic communications via the Paper PDA.

**Keywords:** Hybrid paper electronic interfaces, interaction on paper, augmented reality, interaction techniques, hyperlinking.

## INTRODUCTION AND MOTIVATION

Paper materials have well known merits from a usability standpoint (for a discussion see for example [6,12]). They are familiar, portable, inexpensive, highly adaptable, and have very good readability characteristics [13,14]. In fact, being as natural as “pencil and paper” is a lofty, but typically unobtained, goal for most interactive systems.

---

This work was supported in part by Microsoft Research and in part by the National Science Foundation under grants IRI-9500942 and IIS-9800597.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

*UIST '99, Asheville, NC*

© 1999 ACM 1-58113-075-9/99/11... \$5.00

However, paper is not (by itself) computational in nature, and can't offer a number of the advantages of a computational medium. As a result, there has been a long history of work attempting to create an electronic analog of paper in one form or another. For example, work in handwriting recognition dates at least to the early 1960's (see for example [4]).

Recently this long line of work has born fruit. For a few hundred dollars, one can now buy a small hand held device, such as the popular 3Com Palm Pilot, which is capable of recognizing and acting on hand written input. However despite the tremendous progress that has been made, these devices are still a long way from the convenience, “feel”, and ease of use of, for example, a paperback book.

In order to attempt to remedy this, the work presented here takes another approach. Rather than creating devices to replace paper, it seeks to add various electronic capabilities to actual paper. We use the term *hybrid paper electronic interfaces* to describe such systems. This paper focuses on several interactive techniques and applications that operate within the framework of one such system – the *Paper PDA*.

One particularly successful use of paper for an information rich task is the use of calendars, appointment books, or organizers for personal information. Overall, because they are highly useable and quite functional, these paper “systems” have gained widespread user acceptance – typical office supply stores now carry a wide array of products designed to work with this type of small paper notebook.

Recently, *Personal Digital Assistants* (PDAs) have attempted to replace traditional paper organizers. Like other replacements for paper, these electronic systems offer a number of important advantages because they are computational devices. However, while more capable, these systems are not generally as natural to use, nor as flexible as a typical paper organizer.

Work on the Paper PDA project looks at ways, not to replace a paper organizer with a PDA, but instead to augment real paper with PDA-like functionality. Users carry a notebook filled with ordinary paper and write on it with an ordinary pen – hence the system maintains the highly desirable usability properties of real paper. However, the printed contents of that notebook comes (partially) from an electronic system, and marks made in

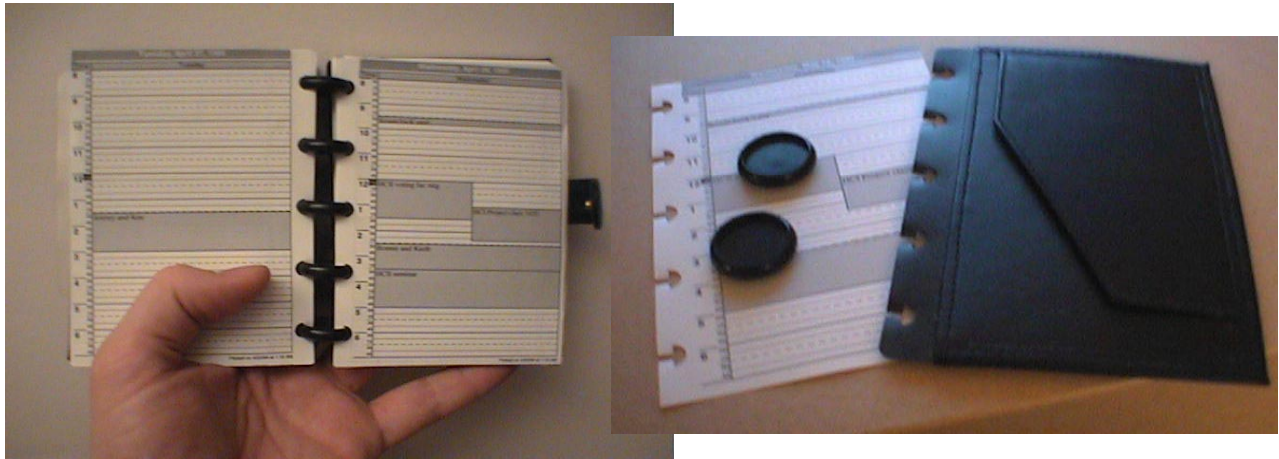


Figure 1. The Circa Modular Notebook System Used for Prototyping

the notebook have consequence in the electronic world – hence the user gains some of the benefits of a computational medium.

The Paper PDA has its roots in a simple observation. The highly successful 3Com Palm Pilot operates in a mostly disconnected fashion, with only periodic "synching" with a larger (and perhaps networked) computer. A similar mode of operation can be created with a paper organizer. A paper notebook can be "synched" with an electronic representation by scanning (and partially recognizing and processing) its contents, then printing a new copy.

By periodically recreating a fresh copy of the printed contents, items coming from the electronic world can be integrated into the notebook. For example, entries from a group scheduling system can be made to appear in individual organizers, a display of "available times" for group members can be displayed, etc. Further, because of its electronic nature, written material in the notebook can potentially be moved around and reorganized (at the next printing) into different layouts to suit different amounts of content, or different tasks. While not fully dynamic like a pure electronic display, this *semi-dynamic* representation is still considerably more versatile than a pure paper system.

In addition, because the basic framework and detailed layout of each paper page is controlled by the system, a number of useful additions can be made to the printed page which facilitate extracting content from the paper and allowing marks on paper to invoke actions in the electronic world.

For example, as described later, registration marks can be added to each page to improve scanning and OCR (by providing precise rotation and orientation information). Further each page, and even each version of each page, can be assigned a unique identifier. This identifier can be placed on the page in an electronically recoverable form. Since the layout of each page is known to the system and can be indexed by its unique identifier, this allows easy segmentation of the page image (e.g., extraction of marks from particular fields). In general, this allows the creation

of on-paper interactive capabilities whereby affordances such as check boxes and text entry fields can be placed on the paper. Marks within these special locations can be easily located and processed in order to carry out actions in the electronic world. (This capability was pioneered in the XAX [8] and Protofoil systems [15], which were important inspirations for this work.)

The overall framework of the Paper PDA system offers many new possibilities for hybrid paper electronic interfaces and our work on it is currently at an early stage. This paper concentrates on several particular interaction techniques. However, the detailed usability of these techniques has not been studied, nor is the overall Paper PDA system ready for widespread use. For example, the difficulty of synchronizing (unbinding, scanning, reprinting, and rebinding a notebook) is critical to the eventual usability of the system. However, this step is currently done in a manual fashion which is still more cumbersome than we would like. Nonetheless, the current prototype has allowed us to open up exploration of a rich new space of interactive techniques.

In the next section the general operation of the Paper PDA and its current prototype will be discussed. Basic infrastructure for the system will be considered next, followed by the details of a paper-based messaging application. This will be followed by discussion of the StickerLink technique, then another messaging application. Finally, conclusions and future work will be considered.

#### OPERATION OF THE PAPER PDA PROTOTYPE

To facilitate easy assembly of notebooks and experimentation with different physical forms, the current Paper PDA prototype makes use of a modular notebook system – the *Circa* system sold by the Levenger Company [9]. This system employs a series of disks along with paper punched with specially shaped holes [5] as shown in Figure 1. These disks hold the paper firmly in place, but also allow pages to be inserted and removed as needed. In addition, because of the close and regular spacing of the punched holes, pages and other artifacts (such as dividers, bookmarks, and pockets) of different sizes can be

accommodated in a single notebook. We are currently in the early stages of experimenting with the form and layout of the prototype. However, we expect that no single form or layout will be right for all users or all tasks. Instead we are concentrating on creating a system which can be adapted to many different forms.

Each page in the Paper PDA is printed with two primary registration marks. Upon scanning of a page, the first processing step is to find these registration marks (see below). We expect to use several different form factors and major layout variations for pages. To accommodate this, we support a set of *page families*. Each page family is associated with a type and placement of registration marks that is uniquely identifiable. Once the centers of the registration marks are established, and the page family determined, the page image is transformed to remove any rotation caused by misalignment of the page in the scanner. Note this eliminates a “de-skewing” step in a typical OCR process (see for example [1,3,10]), and can be performed in a more robust and reliable fashion than is possible without registration marks. Eliminating this step will be particularly important for the StickerLink technique described later.

Each page is assigned a *page identifier* that is unique to that particular printing of the page. Associated with each page family is the location and orientation of its unique identifier relative to the registration marks. Note that if the visual designer creating a new page layout finds the location of the page identifier to be inconvenient, or aesthetically unattractive, the identifier can be placed in a different location. The designer need only use a new set of registration marks and create a corresponding new page family. Page identifiers can either be in purely machine readable form (e.g., a bar code, or a more aesthetically pleasing *glyph* encoding [8]), or as in our examples, can be textual. Textual identifiers are recovered by OCR. High accuracy of OCR can be expected in this case since the system prints the text, can control the font used, and may manipulate the dictionary used by the OCR system to include (only) currently active ID strings.

After family identification, the image of the page identifier is passed to the identifier decoder routine for the page family. Once a page identifier has been decoded, it is used to look up a page handling routine in a table of page handling methods. The rotation corrected image for the page is then passed to that handler for processing. In this way the Paper PDA system is easily extended with new types of page processing – potentially without even recompiling the system. Each of the applications described below is embodied in a particular handler routine. However, as described in the next section, there are a number of basic capabilities that are common across applications.

### BASIC CAPABILITIES

The first basic capability shared across the system is the location and decoding of registration marks. These marks,



Figure 2. Example Registration Marks and Corresponding Radial Profiles

as illustrated in Figure 2, are radially symmetric patterns — bull’s eye targets with varying ring sizes and grayscale intensities. This type of pattern was selected to ensure that the registration mark can be recognized as such regardless of its rotation.

To find registration marks the system makes a sweep over a newly scanned image, testing the neighborhood around each pixel to see how symmetric it is. It does this by calculating the variance in intensity of the pixels equidistant from the center pixel of the neighborhood. If the variance is low within a given radius, then the center pixel is potentially the center of a registration mark.

The variance for a valid mark is unlikely to be zero because of small imperfections introduced during the scanning process. Further, by allowing slightly higher variances the system can be made to recognize nearly symmetric patterns as registration marks. The freedom to be slightly non-symmetrical can be important in allowing a visual designer to integrate registration marks with other content in visually appealing ways. For example, we make use of the “Q” logo seen in Figures 3 and 6 (below) as a “hidden” registration mark. Similarly, in Figure 6, the second registration mark is fully integrated into the design of page.

As a neighborhood is tested in increasing distance from a point, the system also builds up an intensity profile – a list of the intensities at increasing radii. Uniformly colored areas above a certain size (which are trivially symmetric) can be easily recognized by their profiles and eliminated from consideration. By comparing profiles, the system can distinguish between different registration marks, and further eliminate false positives. The *type* of a registration mark – used for example to identify a page family – is determined by matching detected profiles against the set of known registration mark profiles.

To make the creation of new kinds of registration marks easy and accessible to visual designers, we allow them to be created in any convenient drawing program. In order to acquire the necessary radial profile data, a set of sample marks are scanned and “manually” identified in the scanned image as described below.

A second major shared capability is for *paper widget* specification and processing. Some of the information in the Paper PDA may well be maintained as simple free form strokes (stored as an image). However, a typical Paper PDA application will make at least some use of paper

widgets in order to allow marks made on paper to invoke actions in the electronic world.

A paper widget appears to the user as a printed affordance that provides a place to enter or edit information. It carries out actions when marks over or inside that affordance are detected in a scanned image – specifically when marks appear in the *active area(s)* of the widget. Like widgets in a typical GUI toolkit, each paper widget is implemented by an object. This object has several capabilities. These include producing or augmenting a printed image of the widget’s affordance, detection, analysis, and recognition of marks over it in a scanned image, and acting upon those marks. As in a typical GUI toolkit, most widgets will support arbitrary actions via registered callback routines. This allows a general library of widgets to be created and shared, while still allowing widgets to be used in very application specific ways.

In order to make the creation of new page layouts using paper widgets very accessible to visual designers, collections of paper widgets are placed into *page layout templates*. The appearance of a template can be created in any convenient drawing program. Images of these templates are then either scanned from a paper copy or captured directly in electronic form. A small interactive program then allows the designer to designate the location of reference marks (creating a new page family if necessary), to establish the bounds and/or active areas of various paper widgets, and to indicate areas where particular widgets are to augment the template output with printed data.

Pages that make use of paper widgets can all employ a common widget handler routine. This routine is passed the rotation corrected image of the page, along with a page layout template, and a collection of paper widget instances. Partial images are extracted based on the geometry information in the template. These partial images are then passed to the corresponding paper widget objects for processing. The widget normally starts with a mark detection phase where it determines if relevant marks have been made over it. If marks are detected a recognition phase establishes an interpretation for them. Finally, the widgets then carry out actions as needed – typically via a callback.

In the next sections we will consider several applications and interaction techniques that make use of the basic Paper PDA framework.

### MAILQ: MESSAGING FROM PAPER

One reason email has been so successful is that it is fairly lightweight – one can “just jot down a note” and send it. Small paper notes of course have similar properties, but because they are physical objects, delivering them is not always such a lightweight operation. The *MailQ* application is designed to act as a bridge from quick paper notes into the electronic email infrastructure.

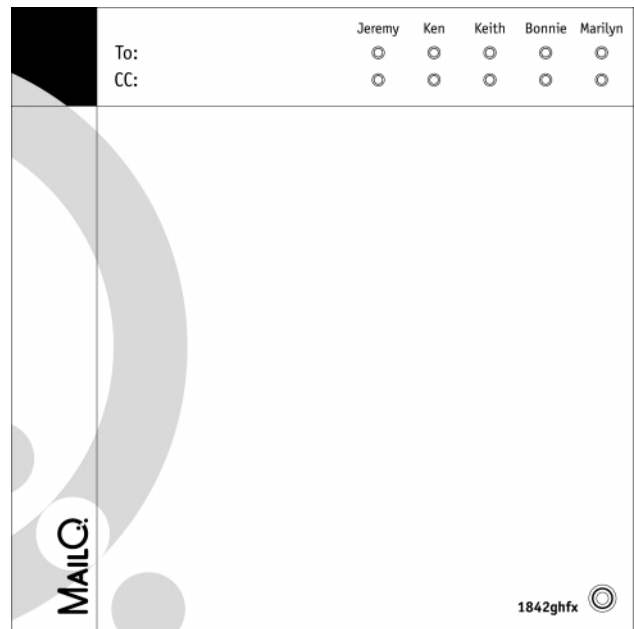


Figure 3. Example MailQ Page

Figure 3 shows a MailQ page. To send a message with MailQ, the user finds a blank MailQ page in their Paper PDA, jots down a note, and then specifies a recipient. The message is processed when the PDA is next synchronized by encapsulating the image of the page in an email message.

Email recipients can be specified in one of several ways. If desired, a small set of common recipients (such as one’s secretary and key colleagues) can be printed on the page with checkboxes. These can be specified simply by checking the appropriate box. While different sets of common addressees can be printed on different pages, only a few common addressees can be handled in this way. For a larger set of addressees, we make use of a *StickerLink*.

As described in the next section, a *StickerLink* is a small removable paper sticker with a machine recoverable (and typically human readable) identifier printed on it. Various objects in the Paper PDA have prepared *StickerLinks* printed with them. To create hyperlinks and other associations between objects in the Paper PDA, a *StickerLink* that refers to one object is placed inside another object. This provides a general paper-to-paper object association or hyperlinking capability which can be used to create corresponding electronic associations or links.

In the case of MailQ addressees, several *StickerLinks* are prepared for each person listed in the user’s address book. One or more of the stickers can be placed on a MailQ page to specify a recipient.

A final method for addressing a MailQ message is to write in the recipient manually. Eventually, this will be done in a set of provided boxes and off-line handwriting recognition will be used to extract the recipient. However, to simplify

the current prototype we have used the temporary measure of allowing a free form entry with human recognition of the address. This is done by forwarding MailQ messages with free form addresses to a fixed designee (typically the user themselves, but possibly their assistant) who manually enters the final address.

While only a temporary measure for the MailQ prototype, this kind of *semi-automatic* approach is generally useful in other places. In particular, it provides a graceful degradation path. It provides a way to allow the Paper PDA to operate in situations where full recognition of the meaning and intent of marks on paper is beyond the current state of the art. Further, it provides a mechanism for error recovery when recognition results are uncertain. (This is a particularly important issue since user action and recognition are now separated in time.)

Although not implemented in the current prototype, in addition to specifying recipients, it is also possible to attach other Paper PDA objects to the body of MailQ messages using StickerLinks. For example, it might be useful to attach a scheduled appointment, an address book entry, or a to do list entry to a message. In order to support linking to arbitrary objects in the Paper PDA (without preparing a sticker for each in advance), a two-part StickerLink can be used. A two-part StickerLink contains two matched

stickers, one placed on the object linked from, and one placed on the object linked to.

Like many Paper PDA applications MailQ is relatively simple to implement once the overall Paper PDA infrastructure is in place. Following the usual method, when the PDA is synchronized, each MailQ page is processed. First, its registration marks are located. The form and placement of these marks indicate the family of the page, and this is used to extract the page identifier (visible in the lower right corner of Figure 3). This identifier is looked up in a page association table which indicates that the page is a MailQ page, hence its image is passed to the MailQ page handler (after rotation correction). The MailQ handler looks for StickerLinks (as described below) and for marks within the "To:" area of the page. It then bundles the image of the page into a mail message with the given recipients. If marks in the "To:" areas were located, the message is encapsulated and forwarded inside another message to the fixed designee for further processing. An example of such a message appears in Figure 4. (Note that this message is formatted using embedded HTML in order to implement a small user interface.) If all recipients have been resolved, then the mail message is simply sent to those recipients without human intervention.

### STICKERLINKS

As described above, StickerLinks provide a physical (paper-to-paper) linking mechanism which can also create analogous links within the corresponding electronic material. (A conceptually related paper-to-electronic linking system using printed Post-It notes for web page bookmarks is described in [11].) This form of link can be used in a number of ways to enhance the capabilities of the Paper PDA. In particular, StickerLinks can be used to make the otherwise static medium of printing on paper more flexible and dynamic.

Two example uses of StickerLinks were presented above which were oriented towards object association and data entry. A third potential example illustrates how StickerLinks can support more dynamic interaction on paper. In an electronic calendar it is typically possible to attach notes to calendar appointments. This happens for example on the Palm Pilot by popping up a dialog box for entering or reading extra text. Such a dynamic display is not possible on a static printed page. Further, we typically cannot afford to place extra space with every entry that *might* have a note attached to it. However, we can use StickerLinks to create a paper analog of the separate dialog box.

This can be done by providing a page with blank areas for notes. Each section of this page would then be provided with a StickerLink. To attach a note to a calendar entry (or any other entity) one would turn to the first available note page section, jot down the note, then place the corresponding sticker on the object being annotated. When the PDA was next synchronized, the note would be



Figure 4. Message Requesting Manual Routing of a MailQ Message

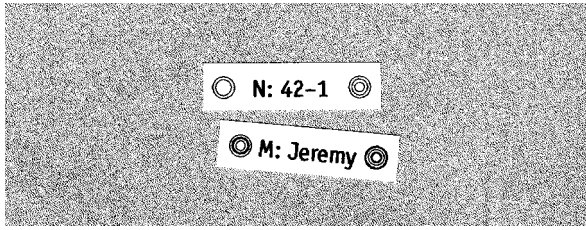


Figure 5. Example StickerLinks

incorporated into the calendar entry. This might for example, involve reformatting the entry to provide space for the note (or part of the note).

To operate properly, StickerLinks need to be identifiable. Not surprisingly, each sticker has an identifier printed on it. Since stickers are placed individually by the user, they may not be aligned with the page they are placed on. As a result, registration marks are particularly important for StickerLinks. Each sticker is printed with two registration marks, one at each end of the sticker identifier. These marks enable the sticker to be discovered easily, for the sticker images to be extracted from the overall image, and for that image to be accurately rotation corrected. In addition, the pair of registration marks encodes a small amount of information that is used to identify the type of StickerLink.

Note that until the PDA has been synchronized, the StickerLink needs to serve as a paper-to-paper hyperlink that can be followed manually. Because of this need, it is extremely useful for the identifying information on the StickerLink to be human readable. To accomplish this we use textual strings for identification and recover them through OCR (of the segmented and rotation corrected image).

In the case of an attached note, the printed identifier would typically be an identifying prefix followed by a page number, and then a section number within that page (e.g., "N:37-2"). For addressees in the MailQ application, the identifier would be an identifying prefix followed by the unique part of a recipient's name (e.g., "M:Jeremy"). Example StickerLinks are shown in Figure 5.

### QCARDS

A final application example for the Paper PDA is *QCards*. Like MailQ this application is concerned with communicating electronic information via paper artifacts. However, in this case we are interested in creating useful (and sharable) paper representations of electronic information, rather than creating an electronic entity from a paper one.

The QCards application is motivated by a typical information sharing scenario that occurs frequently at technical conferences. In that setting one often wants to share a document or other piece of information (such as a published paper, a description of a research project, or a piece of code) with a colleague. However, even if one could anticipate which information was needed, its

normally too cumbersome to carry around a collection of documents that one might want to give out. Instead, a typical ad hoc procedure for doing this involves taking the business card of the intended recipient, and writing a note to yourself on the back. Once the conference is over, the information can be emailed to the recipient.

This interaction has the important property that it is very lightweight in nature, and can be done without unduly disrupting a natural conversation (compare this, for example, with getting out a palmtop device, entering an email program, transcribing the recipient's email address, and sending oneself email). Unfortunately, this process is rather error prone in practice – we often forget to send the information after returning.

The QCard application allows a similar transfer of information to happen in a similarly lightweight fashion, but is capable of automating some or all of the steps, making the transfer easier (and hence more likely to be completed). QCards work with a two part perforated card as shown in Figure 6. The right portion of the card is the size of a standard business card (and in fact is printed on standard perforated business card stock). That part of the card is given to the recipient of the electronic information and serves as a physical token representing that information [2]. The left hand side of the card is retained in the Paper PDA and scanned at the next synchronization.

The right hand ("take away") part of the card contains a unique URL which matches the page identifier found on the left hand ("kept") part of the card. The receiver of the information can go to that URL to access the information. The left hand part of the card contains a series of check boxes indicating information one anticipates possibly

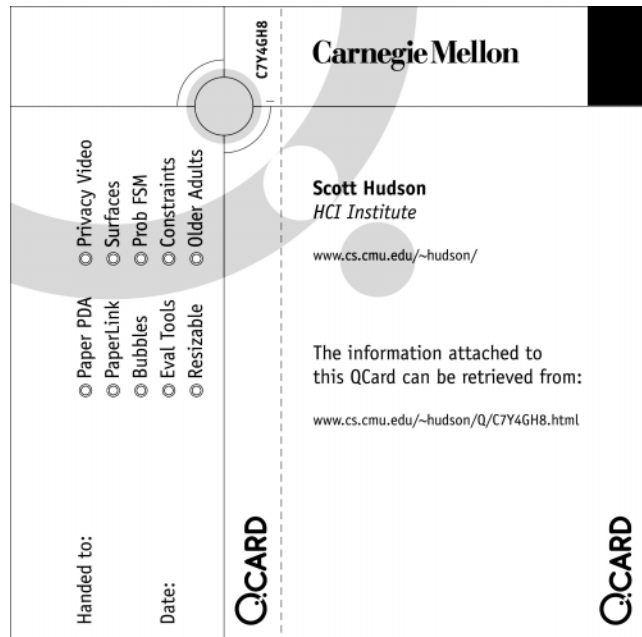


Figure 6. Sample QCard

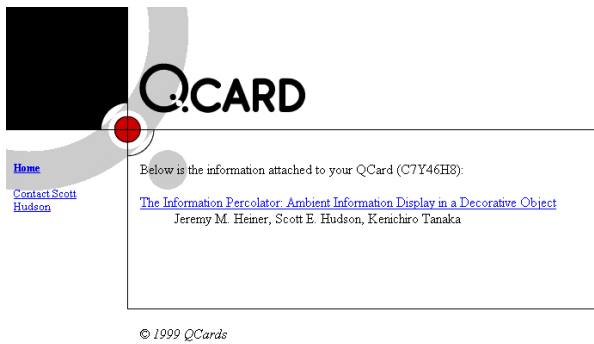


Figure 7. Sample QCard Information Access Web Page



Figure 8. Sample QCard "Not Loaded" Web Page

giving out, along with a free-form area for a note describing things which were not directly anticipated.

When the kept portion of the card is processed, any cards with only check box marks are handled in a fully automatic fashion. In that case, the system prepares a new web page similar to the one shown in Figure 7 and links it to the requested information. If a free form note has been included, then the image of the card is packaged in a mail message back to the issuer of the card as a reminder to place the information. Currently this is a mail message with a simple image attachment. However, in the future we intend to place a small Java or JavaScript program in the message which automates selection and placement of information (for example by providing a file pick interface and doing the corresponding placement of the file in web space automatically).

Until cards have had content assigned to them in web space a placeholder web page similar to the one shown in Figure 8 is provided. To help complete the placement of information, this page has a button which, if pressed, will send a reminder to the card owner (this button can also be left out at the discretion of the owner).

The QCard application is implemented in much the same way as the MailQ application, except that its actions involve creation of web pages in addition to sending of email messages.

In design of the system we were initially concerned about garbage collecting the web pages associated with QCards.

For example, if a paper QCard were lost after being issued, we would not know that the information was now inaccessible, nor in general would we know when it was safe to throw a QCard web page away. While searching for a solution to this dilemma we realized, however, that our intuition about this problem was leading us astray. At the time of this writing, a 10 Gbyte disk drive can be purchased for under \$200 (not including power supply or interface). If we assume pessimistically that each QCard has 500 Kbytes of unique storage associated with it\*, the cost of storage for each such page is under \$0.01. Even if we triple this cost to account for power supply and interface, this means that the paper QCard (which we had considered disposable) probably now costs more to produce than storing the electronic information it represents.

Based on this analysis we have adopted a nominal policy of keeping QCards for the life of the disk drive they are stored on. As a part of future work, individual QCards may be given shorter (timed) lifetimes via checkboxes on the kept portion of the card. It might also be possible to allow some QCard pages to be marked for deletion by the person receiving the information.

#### CONCLUSION AND FUTURE WORK

This paper has considered the basic structure and operation of the Paper PDA system along with several interaction techniques and applications for use with it. Because interaction with the PDA makes use of a real paper notebook and marks from an ordinary pen, we believe it has much potential for maintaining the naturalness, adaptability, and general ease of use of paper. On the other hand, because it integrates electronic information and capabilities with those of printed materials, we believe the system also has the potential for going significantly beyond the traditional limitations of paper.

As outlined in this paper, the Paper PDA system provides a general and conceptually powerful framework that can be expanded with a number of different applications and interaction techniques. This paper has considered a few preliminary examples of these. However, work on this system is in the early stages and we believe there is considerable potential for future work.

In particular, the area of paper widget design is largely unexplored. This area is challenging because, while quite elaborate affordances for interaction can be printed on the page, no real dynamic feedback can be provided. As an example, interaction techniques for deletion and revision of entered information need to be developed. Similarly, provisions for undo and redo in the semi-static world of paper are needed. In addition, new compact techniques for application specific data entry would be useful. For

\*Without linked information or shared image files actual QCard information web pages take up about 16 Kbytes each and we expect most linked information to be shared between several cards.

example, entry of information specifying how a repeating appointment should be repeated, and when the repeating series ends, will be needed.

One general approach to these new techniques may be to use sophisticated off-line gesture and handwriting recognition techniques. However, it may also be useful to instead consider techniques which minimize and simplify the required recognition in order to minimize the uncertainty that is a normal part of recognition based systems.

Because at least some recognition will be involved, another generally important area for future work is the handling of errors and uncertainty. This is particularly important since, in the Paper PDA system, the time of user input is separated from the time of system action. A general strategy that we intend to continue using is to employ semi-automatic or "human in the loop" techniques [7] where more difficult recognition tasks may be performed by humans, or where the user is called upon to monitor the operation of the system when uncertainty is high. A significant challenge with this approach is to make it "lightweight" enough that it does not significantly detract from the usability of the overall system.

A final important area for future work will be in the design of the physical form of the paper portions of the system (such as the form factor, appearance, and "feel" of the notebook used). Even more critical will be the mechanism used for the synchronization (unbinding, scanning, printing, and rebinding) process. If this process is too cumbersome, regular synchronization of the PDA will be impeded making it generally less connected to the electronic world.

## REFERENCES

- [1] Baird, H., "The skew angle of printed documents", *Proceedings of the SPSE 40th Annual Conference and Symposium on Hybrid Imaging Systems*, 1987, pp. 21-24.
- [2] Barrett R., Maglio, P., "Informative Things: How to Attach Information to the Real World", *Proceedings of the ACM Symposium on User Interface Software and Technology*, 1998, pp.81-88.
- [3] Bessho, G., Ejiri, K., Cullen, J., "Fast and accurate skew detection algorithm for a text document or a document with straight lines", *Proceedings of the SPIE - Document Recognition*, v2181, 1994, pp. 133-140.
- [4] Earnest, L., "Machine Recognition of Cursive Writing", *Proceedings Of IFTP Congress '62*, Munich, Germany, Aug-Sept 1962, pp. 462-466.
- [5] Feldman, J., Czuckerberg, S., "Notebook System", US Patent 5553959, Sept. 1996.
- [6] Harper, R., Sellen, A., "Collaborative Tools and the Practicalities of Professional Work at the International Monetary Fund", *Proceedings of CHI'95*, 1995, pp.122-129.
- [7] Hudson, S., Hsi, C., "A Synergistic Approach to Specifying Simple Number Independent Layouts by Example", *Proceedings of INTERCHI'93*, 1993, pp. 285-292.
- [8] Johnson, W., Jellinek, H., Klotz, L., Rao, R., Card, S., "Bridging the Paper and Electronic Worlds: The Paper User Interface", *Proceedings of INTERCHI'93*, 1993, pp. 507-512.
- [9] Levenger Inc., "Levenger Company Home Page", Web site: <http://www.levenger.com/>.
- [10] Liu, J., Lee, C., Shu, R., "An efficient method for the skew normalization of a document image", *Proceedings of the International Conference on Pattern Recognition*, v III, 1992, pp. 122-125.
- [11] Ljungstrand, P., Holmquist, L., "WebStickers: Using Physical Objects as WWW Bookmarks", *Conference Companion of CHI*, 1999, pp.332-333.
- [12] Mackay, W., Pagani, D., Faber, L., Inwood, B., Launiainen, P. Brenta, L., Pouzol, V. "Ariel: Augmenting Paper Engineering Drawings", *Conference companion on Human Factors in Computing Systems*, 1995, pp. 421-422.
- [13] Muter, P., Latremouille, S. A., Treunit, W. C., & Beam, P. "Extended reading of continuous text on television screens", *Human Factors*, 24, 1982, pp. 501-508.
- [14] O'Hara, K., Sellen, A., "A Comparison of Reading Paper and On-Line Documents", *Proceedings of CHI '97*, 1997, pp.335-342.
- [15] Rao, R., Card, S., Johnson, W., Klotz, L., Trigg, R., "Protofoil: Storing and Finding the Information Worker's Paper Documents in an Electronic File Cabinet", *Proceedings of CHI'94*, 180-185, 1994.